LINE-OF-SIGHT STABILIZATION AND TRACKING CONTROL FOR INERTIAL PLATFORMS

Matheus Ferreira dos Reis

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia Elétrica, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia Elétrica.

Orientadores: Ramon Romankevicius Costa
Antonio Candea Leite

Rio de Janeiro

Agosto de 2018

LINE-OF-SIGHT STABILIZATION AND TRACKING CONTROL FOR INERTIAL PLATFORMS

Matheus Ferreira dos Reis

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA ELÉTRICA.

Examinada por:

Prof. Ramon Romankevicius Costa, D.Sc.

Prof. Antonio Candea Leite, D.Sc.

Prof. Paulo César Pellanda, Dr. ENSAE

Prof. José Paulo Vilela Soares da Cunha, D.Sc.

RIO DE JANEIRO, RJ – BRASIL

AGOSTO DE 2018

Reis, Matheus Ferreira dos

Line-Of-Sight Stabilization and Tracking Control for Inertial Platforms/Matheus Ferreira dos Reis. – Rio de Janeiro: UFRJ/COPPE, 2018.

XVII, 137 p.: il.; 29, 7cm.

Orientadores: Ramon Romankevicius Costa

Antonio Candea Leite

Dissertação (mestrado) – UFRJ/COPPE/Programa de Engenharia Elétrica, 2018.

Referências Bibliográficas: p. 116 – 122.

1. Vehicle-Manipulator Systems. 2. LOS Control. 3. Computed Torque Control. 4. Sliding Mode Control. I. Costa, Ramon Romankevicius *et al.* II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Elétrica. III. Título.

A todos que me guiaram pela jornada neste mundo, que é a vida.

Agradecimentos

A caminhada para a conclusão de mais esta etapa em minha vida foi árdua, porém prazerosa e gratificante. Em primeiro lugar, sou eternamente grato à minha querida "pãe" Angela Reis. Não é nenhum exagero dizer que, sem a sua ajuda e suporte incondicional, eu simplesmente não teria chegado até aqui. Obrigado, mãe. Agradeço também à minha inspiração, minha vida e meu amor, Karina Fernandes Merlo, que me acompanha de lá das terras lusitanas e por algum motivo escolheu a mim no meio de tantos outros loucos.

Gostaria de agradecer também a todos os que me acompanharam e auxiliaram durante o percurso de graduação e mestrado, entre eles os professores da Coppe-UFRJ, pela dedicação, conselhos e sugestões. Agradeço especialmente ao meu orientador Ramon Romankevicius Costa, que me deu a oportunidade de trabalhar neste projeto e sempre me apoiou em minhas decisões para o futuro e ao meu coorientador Antônio Candea Leite, pelo incentivo constante, atenção sobrenatural aos detalhes e pela amizade e confiança que deposita em mim. Além deles, agradeço também aos membros da banca examinadora, professores Paulo César Pellanda e José Paulo Vilela, pelas sugestões enriquecedoras a este trabalho.

Agradeço também aos professores Alessandro Jacoud, Fernando C. Lizarralde, Jomar Gozzi, Afonso Celso e Liu Hsu, por terem participado diretamente em minha formação e me incentivado na eterna busca pelo conhecimento. Agradeço aos meus colegas de laboratório Alex F. Neves (Alex... Aleeex!), Guilherme P. Carvalho (Zeca!), João C. Monteiro, Isabela Lima de Oliveira, Rodolpho Ribeiro, Ignácio Ricart, Gabriel Franco e Ricardo Ouvinã, e também aos meus colegas de aulas Jéssica Vieira, Jonathan Fried e Waldez Junior, e a todos os habitantes do LEAD durante estes anos, pelas sugestões, reuniões, conversas, dicas, brincadeiras e por tornarem o ambiente de trabalho um lugar agradável e descontraído.

Por último, agradeço aos soldados desconhecidos; a todas as forças humanas e naturais que tenham contribuído direta e indiretamente para a minha formação e que devido às minhas limitações não foram citadas aqui. Gratidão!

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

CONTROLE PARA ESTABILIZAÇÃO E RASTREAMENTO DA LINHA DE VISADA UTILIZANDO PLATAFORMAS INERCIAIS

Matheus Ferreira dos Reis

Agosto/2018

Orientadores: Ramon Romankevicius Costa
Antonio Candea Leite

Programa: Engenharia Elétrica

Hoje em dia, a estabilização e o rastreamento da linha de visada utilizando plataformas inerciais continuam a constituir desafiadores problemas de engenharia.

Com a crescente demanda por aplicações de alta precisão, técnicas de controle complexas são necessárias para atingir melhor desempenho.

Neste trabalho, modelos cinemáticos e dinâmicos para uma plataforma mecânica de estabilização inercial são apresentados. Tais modelos se baseiam no formalismo para sistemas veículo-manipulator para a modelagem de manipuladores robóticos operando em uma base móvel (veículo). O modelo dinâmico apresentado segue a formulação analítica de Euler-Lagrange e é implementado em simulações numéricas através do método iterativo de Newton-Euler.

Duas estratégias de controle distintas para estabilização e rastreamento são propostas: (i) controle por torque-computado e (ii) controle por modos deslizantes utilizando o recente algoritmo *Super-Twisting* combinado com um observador baseado em modos deslizantes de alta ordem.

Simulações utilizando dados de movimentação de um navio simulado permitem comparar o desempenho dos controladores por torque computado em relação a um tipo comum de controlador linear utilizado na literatura: o P-PI. Além disso, os resultados obtidos para o controle por modos deslizantes permitem concluir que o algoritmo *Super-Twisting* apresenta rejeição ideal a perturbações provenientes do movimento do veículo e também a incertezas paramétricas, resultando em precisão de estabilização de aproximadamente $0{,}8\,mrad$.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

LINE-OF-SIGHT STABILIZATION AND TRACKING CONTROL FOR INERTIAL PLATFORMS

Matheus Ferreira dos Reis

August/2018

Advisors: Ramon Romankevicius Costa
Antonio Candea Leite

Department: Electrical Engineering

Nowadays, line of sight stabilization and tracking using inertially stabilized platforms (ISPs) are still challenging engineering problems.

With a growing demand for high-precision applications, more involved control techniques are necessary to achieve better performance.

In this work, kinematic and dynamic models for a three degrees-of-freedom ISP are presented. These models are based in the vehicle-manipulator system (VMS) framework for modeling of robot manipulators operating in a mobile base (vehicles). The dynamic model follows the Euler-Lagrange formulation and is implemented by numeric simulations using the iterative Newton-Euler method.

Two distinct control strategies for both stabilization and tracking are proposed: (i) computed torque control and (ii) sliding mode control using the recent *Super-Twisting Algorithm* (STA) combined with a *High-Order Sliding Mode Observer* (HOSMO).

Simulations using data from a simulated vessel allow us to compare the performance of the computed torque controllers with respect to the commonly used P-PI controller. Besides, the results obtained for the sliding mode controllers indicate that the *Super-Twisting* algorithm offers ideal robustness to the vehicle motion disturbances and also to parametric uncertainties, resulting in a stabilization precision of approximately $0{,}8\,mrad$.

Contents

List of Figures

List of Tables

List of Symbols

$\mathbb{H}^*$	Set of unitary quaternions, p. 124
$\mathbb{H}_p$	Set of pure quaternions, p. 124
$\mathbb{N}_0$	Set of natural numbers, p. 28
$\mathbf{E}_0$	Vehicle frame, p. 28
$\mathbf{E}_i$	Link frame, p. 28
$\mathbf{E}_w$	Inertial world frame, p. 28
$\mathbf{E}_{\bar{0}}$	Vehicle CG frame, p. 28
$\mathbf{E}_{\bar{i}}$	Link CG frame, p. 28
$\mathbf{E}_c$	Camera frame, p. 28
$\mathbf{E}_s$	Inertial sensor frame, p. 28
$\mathbf{E}_t$	Target frame, p. 28
I_Q	Unitary quaternion, p. 124

List of Abbreviations

CG	Center of Gravity, p. 11
DoF	Degrees-Of-Freedom, p. 2
FIR	Finite Impulse Response, p. 109
FOSM	First Order Sliding Mode, p. 88
FOV	Field-of-View, p. 1
FSM	Fast Steering Mirror, p. 6
HOSMO	High-Order Sliding Mode Observers, p. 24
HOSM	Higher-Order Sliding Mode, p. 91
IMU	Inertial Measurement Unit, p. 9
INS	Inertial Navigation System, p. 9
ISP	Inertially Stabilized Platform, p. 2
LOS	Line-of-Sight, p. 1
NTSM	Non-singular Terminal Sliding Mode, p. 24
RGB	Red, Green, Blue, p. 3
SOSM	Second Order Sliding Mode, p. 91
VMS	Vehicle-Manipulator System, p. 16
VSAT	Very Small Aperture Terminal, p. 12

Chapter 1

A Survey on Line Of Sight Stabilization

A common problem in engineering is the design of a pointing system for a field-of-view (FOV) sensor such as a camera, laser or radar fixed in a mobile platform. The theory of Line-of-Sight (LOS) control, which has been widely studied over the last century, provides the appropriate tools for dealing with this particularly challenging engineering problem. The LOS is defined as an imaginary line drawn between an observer and an object of interest. The *aimpoint* is the vector pointing in the direction in which the observer is actually heading. If a symmetric FOV is defined for the observer, then the aimpoint would be defined as the vector in the center of the FOV. The LOS error between the aimpoint of the sensor and a moving target arises due to both target-to-platform kinematics and platform motion disturbances.

The LOS control problem stands for correctly pointing the aimpoint vector in the LOS direction, with the lowest error as possible. Under this perspective, the simplest way of thinking about LOS control is to reduce it to the problem of controlling the orientation of the observer's aimpoint. Most pointing systems mounted on a mobile platform require some form of motion compensation to stabilize the aimpoint along the LOS. Therefore, the observer (a sensor) is generally installed along with some kind of electromechanical device with some degrees of freedom, so that the aimpoint can be controlled independently of the base motion.

Looking into nature, an interesting example on the use of LOS stabilization is the eye (Fig. 1.1), which is basically a complex sensor device driven by a structure of muscular actuators. For example, in [10], the influence of visual feedback in the stabilization of the horizontal, vertical and torsional eye movements is studied, and in [11], the first complete model of eye-head stabilization based on the coordination of the vestibulocollic reflex (VCR) and the vestibulo-ocular reflex (VOR) is presented

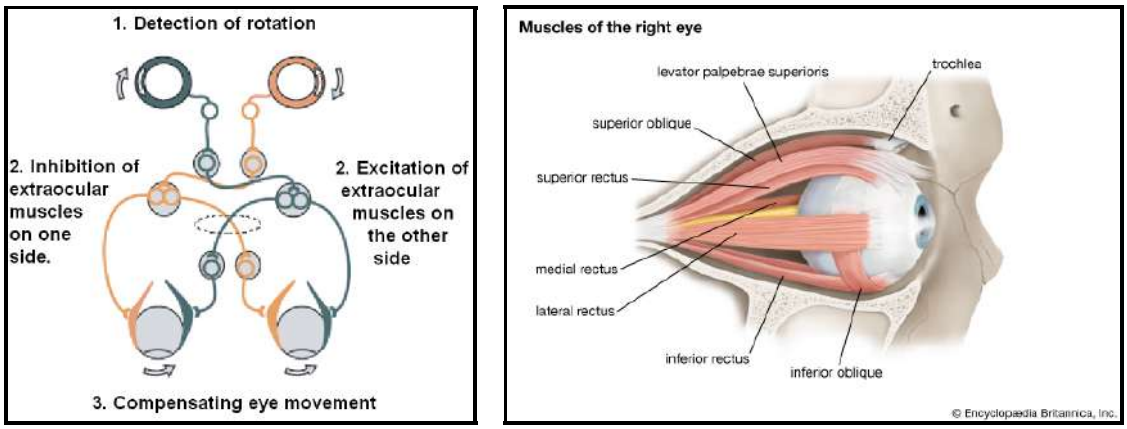


Figure 1.1: Illustration of the vestibulo-ocular reflex, responsible for the horizontal stabilization of the human eye and the internal structure of the human eye. Images were downloaded from https://upload.wikimedia.org/wikipedia/commons/5/58/Simple_vestibulo-ocular_reflex.PNG, https://media1.britannica.com/eb-media/47/63347-004-92824474.jpg.

and tested on a simulated humanoid robot, replicating torso disturbances acquired from a human subject performing several locomotion tasks. The VCR and the VOR reflexes are responsible for the stabilization of the head in space and the visual axis in the image, respectively, both contributing for the overall image stabilization on the retina.

The principle of *mechanical actuation* for achieving a LOS control goal is now present in several technology applications as well, such as image stabilization for cameras, long-range laser pointing, vehicle-mounted radars, robotic applications, precision pointing control for space telescopes and even entertainment industry (Fig. 1.2).

In all applications shown in Fig. 1.2, an electromechanical device is used for controlling the orientation of the object to be stabilized. This device is often called an Inertially Stabilized Platform (ISP), and consists of rotational electromechanical actuators and a housing for the stabilized payload. The actuators provide the necessary Degrees-Of-Freedom (DoF) for the LOS stabilization, although the number of actuators in the system is a design requirement strongly dependent on the application. Mostly, only two or three rotational actuators are necessary. Figure 1.3 illustrates an example of a stabilization problem, where an ISP is used to track a mobile target using a camera mounted on a ship.

As an example of application, [12] presents a real-time video image stabilization system (VISS) primarily developed for aerial robots, combining four independent stabilization layers for: (i) vibration detection via an inertial measurement unit (IMU); (ii) vibration damping by means of mechanical devices; (iii) internal optical

Figure 1.2: Applications of ISPs: image stabilization in UAVs (http://www.uasmagazine.com/uploads/posts/web/2015/04/AES_DaVinci_UAS_14292158319778.jpg), military laser LOS stabilization (http://elbitsystems.com/media/Naval_Laser_800X365.jpg), sensor stabilization for moving vehicles (http://www.leonardodrs.com/media/4643/tua-006.jpg?anchor=center&mode=crop&width=600&height=600&rnd=131387355490000000) and aerospace telescopes (http://www.spacetelescope.org/static/archives/images/screen/hubble_earth_sp01.jpg).

camera image stabilization and (iv) software filters for remaining vibrations. A broad survey on the topic of pointing/polarization alignment for mobile very small aperture terminal (VSAT) operations is covered by [4]. In [2], the topic of LOS stabilization for optical imaging systems (such as thermal, RGB or infrared cameras) is discussed. Lastly, [1] presents the concepts and design of inertial stabilization platforms for a wide variety of applications. Although requirements for ISP designs may vary widely, these all have the common goal of controlling the LOS of some object (usually, a sensor) with respect to some target. The LOS can be the aimpoint of a beam or weapon, the center of the field of view (FOV) of a telescope, or simply the direction to which a sensor must be pointed to.

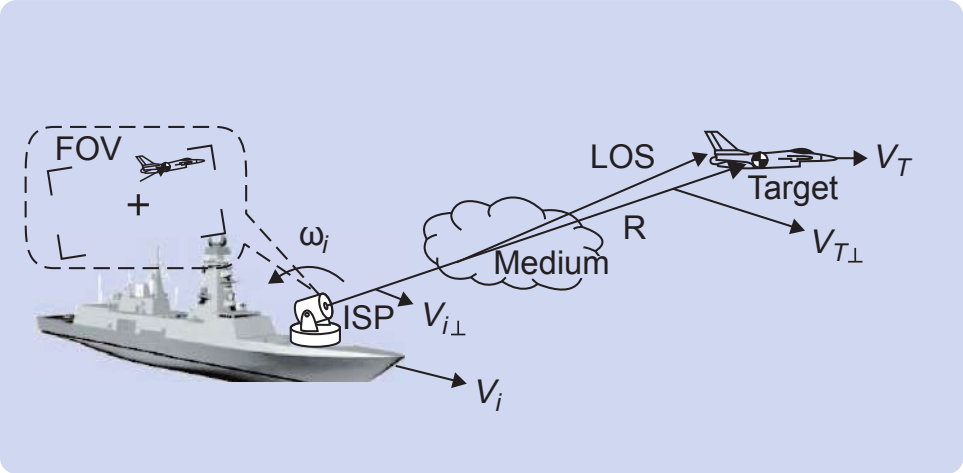


Figure 1.3: Example of a 2DoF stabilization problem, where the target must be always in the camera FOV, without centering the aimpoint to the target. Image from [1].

This chapter is organized in the following way: Section 1.1 discusses the design fundamentals for these mechanisms, from the project requirements to the basic design guidelines. Section 1.2 provides a wide literature review on the theme of mechanical inertial stabilization, while Section 1.3 introduces the problem formulation and the main contributions of this work.

1.1 Design of Inertial Stabilization Systems

Next, we discuss some main topics in ISP design. An ISP must be able to actively reject or sufficiently attenuate the vehicle motion as well as other types of disturbances, such as friction and imbalance on the mechanical parts. Furthermore, generally it must be able to follow a mobile target while rejecting the vehicle motion. The first steps when designing an inertial stabilization system are: (i) understand what are the main concepts and principles used; (ii) identify the system requirements; (iii) study which are the most common electromechanical components and control architectures of an ISP. Here, we focus on the design of inertially stabilized image systems, where the objective is to stabilize the image of a camera.

1.1.1 Requirements

Identifying the stabilization requirements is the first step on LOS control design and it drives many aspects of the system, such as cost, size and weight. The first requirement to be determined is how many DoFs must be stabilized by the inertial platform. In general, these DoFs are referred as roll, pitch and yaw [1]. The majority

of applications require 2-DoFs (azimuth and elevation, or equivalently yaw and pitch), while the roll stabilization is not necessary for LOS pointing. In imaging systems, in order to process the acquired image, the DoF related to the roll motion is usually stabilized or compensated via digital signal processing. However, due to the large roll motion, sometimes it is also necessary to employ mechanical stabilization to guarantee a satisfactory stabilized image.

The other main requirement is precision. For example, for commercial stabilized cameras, stabilization precision is typically in the range of milliradians. Weapon systems and astronomical telescopes, in turn, require extreme precision (microradians or even milliarcseconds) and thus demand much more expensive stabilization systems. The typical performance metric for the precision of the inertial stabilization loop is *jitter*, which is defined as the angular variations of the LOS around a steady state value. For imaging systems, jitter basically quantifies how much the camera shakes due to disturbance residuals after stabilization and is generally specified according to the image blur, which is also related to the camera resolution and shutter speed, besides stabilization performance [2].

Accuracy, in turn, depends strongly on how much systematic error is present in the system, which demands high-accuracy sensors and sufficient knowledge regarding the ISP parameters. A typical performance metric for accuracy is the *tracking error* or *bias* of the tracking control loop, which is the offset of the aimpoint to the LOS.

1.1.2 Platform stabilization and steering stabilization

There are two fundamental approaches for LOS stabilization: *platform stabilization* and *steering stabilization*. Generally, in platform stabilization (or mass stabilization), the entire payload rotates within a gimbal[1] assembly driven by rotational actuators, enabling the aimpoint to be altered relative to the host vehicle (Fig. 1.4). The gimbals are usually mounted one inside of the other, so that each gimbal compensates one platform DoF [1].

For a 2-DoF stabilization problem, for example, the outer gimbal can provide compensation for the yaw orientation, while the inner gimbal provides compensation for pitch. For a 3-DoF problem, the sequence of outer to inner gimbal is usually yaw, pitch, and roll. Robot manipulators with sufficient number of DoFs could also be used for LOS control. In this case, the sensor or payload to be stabilized is generally placed in the manipulator wrist or end-effector, such as in Fig. 1.5, for example. The Doris robot is a mobile robot equipped with a 4-DoF robot arm and a camera in the wrist, for inspection tasks. In this context, it could be used in applications

[1] A gimbal is a pivoted support that allows the rotation of an object about a single axis.

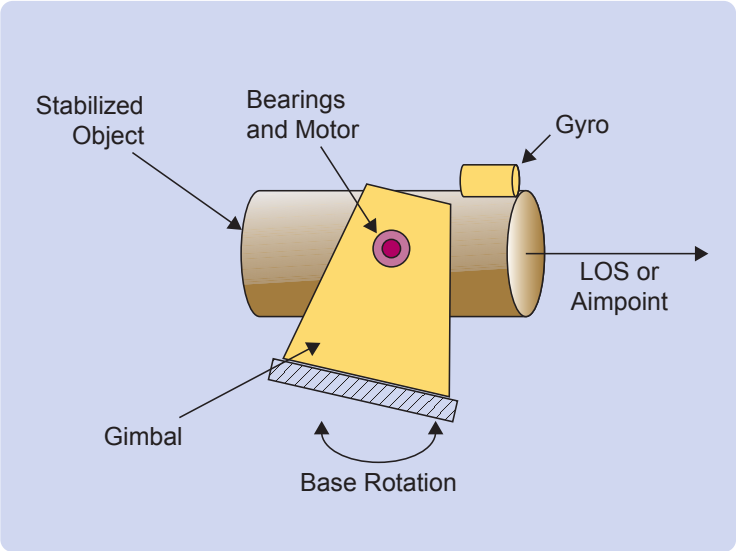


Figure 1.4: 1-DoF gimbal mechanism of platform stabilization. Image from [1].

Figure 1.5: Doris Project, a mobile robot for offshore inspection developed by GSCAR, Petrobras and Statoil.

that require the image to be stabilized while the robot itself moves on the rail. In this case, the manipulator should try to cancel out the disturbances generated by the robot motion.

Another approach of LOS stabilization is named *steering stabilization*, which uses fast steering mirrors (FSM). Instead of moving the entire payload through gimbals to align the aimpoint with the LOS, the sensors are fixed on the host vehicle and gimbaled mirrors are placed on the optical path between the observer and the target (Fig. 1.6). This approach is generally used when the payload to be stabilized is heavy, large or too much expensive to use in a moving platform. In addition, steering stabilization results in a great reduction of size and weight of

the stabilization system. However, this approach can be much more complex, as the stabilization system has to deal, for example, with optical phenomena such as image magnification. Some of these topics can be found in [13].

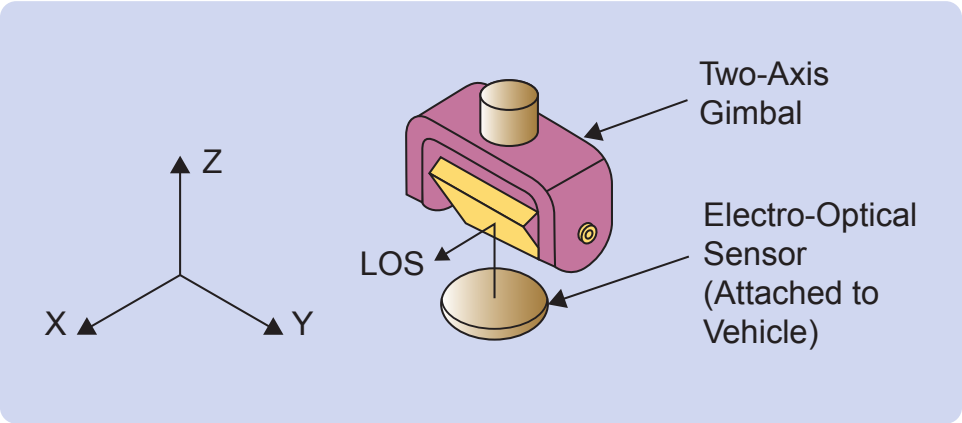


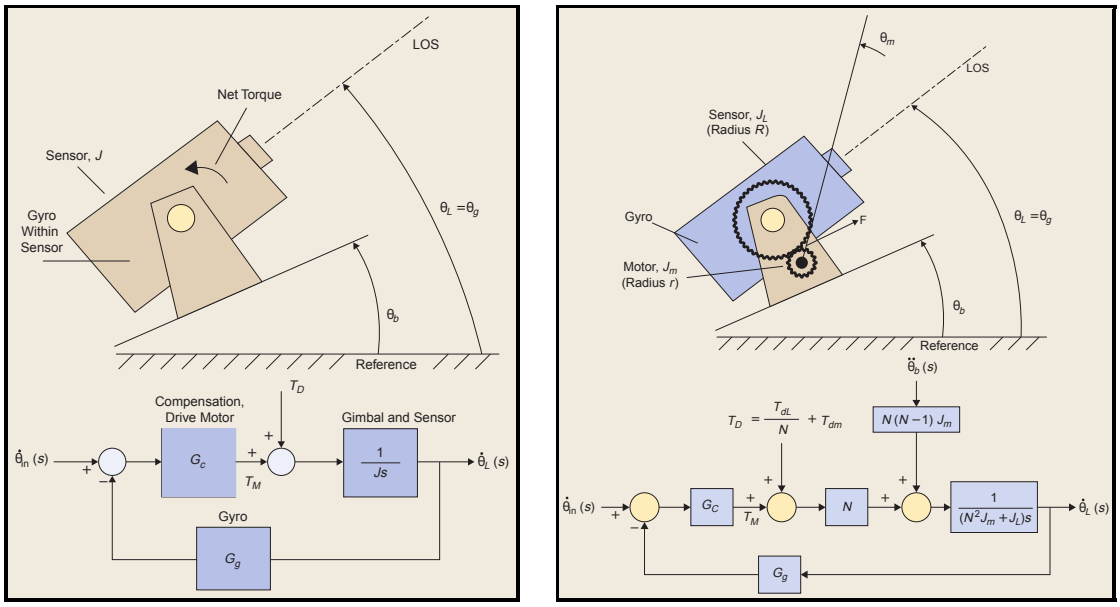
Figure 1.6: 2-DoF stabilization system with steering mirrors. Image from [1].

1.1.3 Direct drive and geared drive

Some of the main components of an ISP are its actuators. According to the literature, the ideal mechanical configuration for mass stabilization is a perfectly balanced mass in gimbals driven by direct drive actuators without friction, cable flexure, or additional disturbances. In this ideal case, the mass remains oriented in its initial position as a consequence of Newton's laws, and the control and dynamic model can be simplified or neglected [2].

A design using direct drive motors allows free rotation of the stabilized mass on a bearing interface, so that the mass inertia aids the stabilization, especially at high frequency disturbance situations. With a direct drive, the load is effectively decoupled mechanically from the motor (Fig. 1.7a). This advantage is lost with a geared actuator, as the gears are coupled to the motor and thus inherently couples the host vehicle motion to the payload, which translates to an additional disturbance to be compensated, as illustrated in Fig. 1.7b. In this case, the motor actuator must accelerate to null the associated LOS deviation, and, consequently, even a frictionless system with gear drives must compensate for LOS motion introduced by the host vehicle pitch, yaw, and roll motion [1, 2, 14].

To achieve high torques without the use of gears, direct drive motors are designed with a large number of poles and a high volume of copper to attain a high torque to power ratio for a relatively low input power. A direct drive motor also provides the greatest practical torque to inertia ratio at the load, making them well suited for applications requiring high accelerations but low speeds. These motors have a



(a) Direct drive gimbal assembly.

(b) Geared drive gimbal assembly.

Figure 1.7: Comparison between direct and geared drive for a 1-DoF system. Note the additional term to be compensated on the control block diagram of the geared case. Image from [2].

small electrical time constant (less than a millisecond), which allows torque to be developed rapidly. Furthermore, the torque varies linearly with the input current and there is almost no magnetic cogging, making these motors very simple to control [1, 14].

Another great advantage of direct drives over geared drives is that these are geometrically suitable for stabilization systems. They usually have a pancake geometry, with a large diameter and narrow width, and also a large hollow shaft, which is appropriate for cable routing. Conversely, gear drives normally present backlash and wear, which are characteristic of gear teeth, and also introduce additional friction to the system [1, 14].

Finally, it is well known that high torque applications would require large direct drive motors, which would be quite expensive, heavy, may require custom fabrication, and would be unfeasible when there are size constraints. Geared drives, however, can provide high torques with a small assembly and have the advantage that the load torque reflected to the motor drive is reduced by a factor equal to the gear ratio. Thus, geared drives are more appropriate when one has to reduce size and weight of the system, particularly when the torque requirements are demanding [1, 2, 14].

1.1.4 Direct stabilization and indirect stabilization

Generally, an important component of a stabilization system is the *inertial sensor*, which measures the kinematic variables of a body relative to an inertial reference, such as orientation, angular velocity and/or acceleration. Since this sensor provides the fundamental measurement of LOS, its installation position is of great importance and may have significant impact on pointing performance. Essentially, there are two strategies to place the inertial sensor: attached to the stabilized mass or attached to the host vehicle [1, 2].

When the inertial sensor is attached to the payload, the stabilization problem is formulated as a *direct LOS stabilization* problem, as shown in Fig. 1.8. The sensor directly measures the LOS disturbances relative to an inertial reference and the stabilization control problem is of nullifying/attenuating the sensor deviations through a high gain servo loop. In this configuration, the control technique usually requires only 2 to 3-DoF gyros mounted on the pointing axes (depending on the number of DoFs to stabilize) [14]. Therefore, *feedback* control techniques are suitable for this type of sensor arrangement.

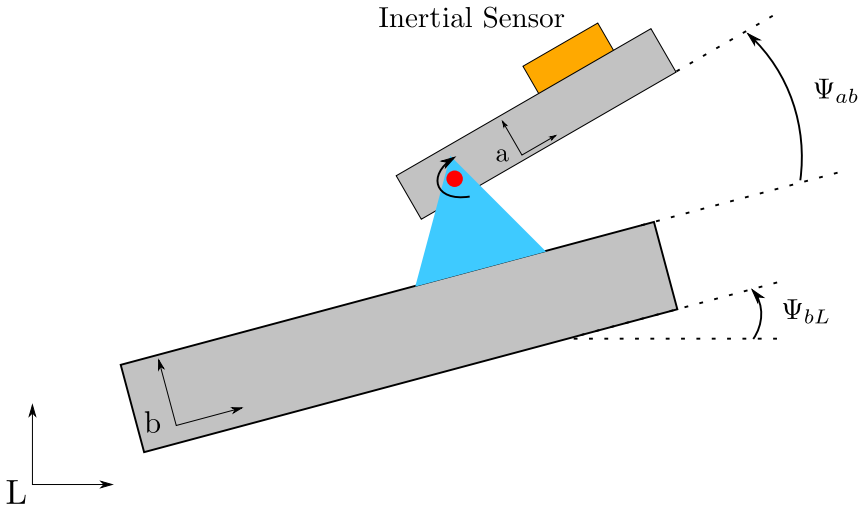


Figure 1.8: 1-DoF direct stabilization. The inertial sensor is attached to the stabilized mass. Image from [3].

The *indirect LOS stabilization* approach, also referred to as strapdown or feedforward configuration [1], consists on fixing the inertial sensors on the vehicle (Fig. 1.9). This requires the measurement of all the three orientations of the vehicle (roll, pitch and yaw), and usually, an Inertial Measurement Unit (IMU) or even an Inertial Navigation System (INS) for precision pointing applications is employed. The system also requires the measurement of the gimbals angles and angular rates, through relative motion sensors (encoders and resolvers), to reconstruct the aimpoint orientation using the host vehicle orientations and the gimbals angles [14]. Therefore, it

is common to apply *feedforward* control techniques for the stabilization.

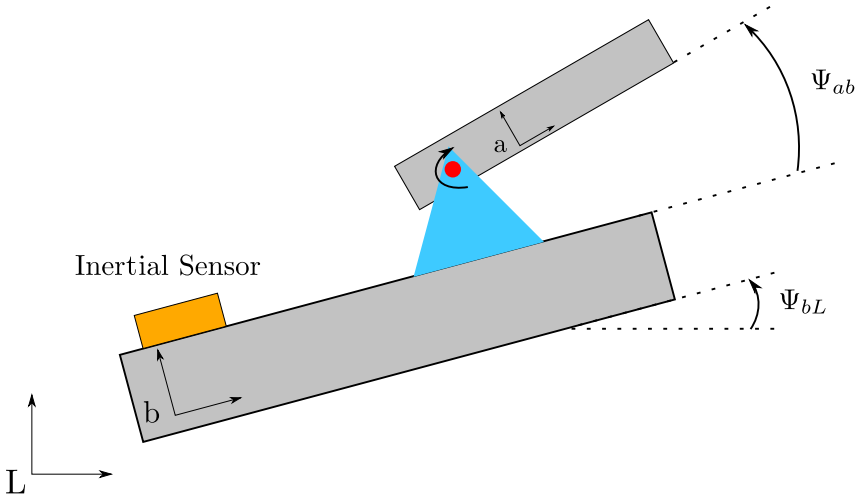


Figure 1.9: 1-DoF indirect stabilization illustration. Image from [3].

The control strategy is different in the indirect case: it does not try to nullify the inertial sensor deviations, but guarantee that the gimbals have opposite motion to these measurements to cancel the vehicle motion effect. Now, the feedback control loop uses angular information provided by the encoders/resolvers besides the inertial sensors. A comparison about the two strategies reveal that the loop gain is equivalent, with the major difference in the noise terms. Whereas the direct approach contains only the inertial sensor noise, the indirect method is also perturbed by the encoder/resolvers noise, and also by the propagation of errors due to the imperfect modeling of the gimbals. Furthermore, phase delay may occur between the two sensor measurements [1].

While the direct strategy suppresses the disturbances by high gain attenuation, the indirect approach requires a cancellation of the measured base motion. Besides the inertial sensors, the encoders/resolvers also must have equally good accuracy and resolution, raising the cost of the stabilization system. Furthermore, the indirect method requires more complex algorithms than the direct approach [1].

Given sensors with equivalent performance specifications, direct LOS stabilization is normally recommended for precision pointing applications [15]. Compensation is derived directly from the measured LOS disturbances, while the indirect approach does not measure disturbances that acts on the ISP, such as air stream disturbance and gimbal structural flexibility.

Gimbal size is impacted by the direct approach, since a larger payload volume is required to mount the sensors on the inner axis of the gimbal. However, it will not always be feasible to place inertial sensors inside the stabilization platform due to size and weight restrictions, especially for mirror-stabilized mechanisms. In addition,

indirect LOS stabilization is improving with the advances of inertial rate and angular position sensor technology in accuracy, bandwidth and data rate, so that direct and indirect approaches are starting to become equivalent. [1, 14].

An alternative configuration for precision pointing applications is known as *strong stabilization* [16]. This method combines direct and indirect stabilization, with gyros mounted on both the sensor frame and on the vehicle. In general, one of the gyros acts as a coarse sensor while the other provides fine residual motion compensation. However, a detailed analysis must be performed to balance the benefits of redundancy and more measurement availability for sensor fusion against the disadvantages of higher cost, weight and noise introduced into the system.

1.1.5 Summary on ISP design

A state of the art ISP typically has a precision of microradians, depending on the application and disturbance magnitude [2]. To achieve this performance, the system has to be carefully designed, including the correct choice of sensors and actuators, as well as precise definition of stabilization requirements, such as the required performance, how many DoFs must be stabilized, and the conditions of the operation environment. A summary containing the general *recommendations* of the literature regarding ISP design are listed as follows [1, 2, 14]:

- platform (or mass) stabilization should be used instead of steering stabilization, when size constraints allow,
- the optimal design specification consists in gimbals with perpendicular pairs of axes, with a common point passing through the ISP Center of Gravity (CG); this ideal condition ensures that the system is not affected by gravity torques and the gimbal dynamics have no cross-coupling effects,
- better accuracy is expected in direct stabilization configurations, since the LOS angles are directly measured; in indirect stabilization, the LOS variables usually have to be estimated by propagating the measured variables through the ISP kinematic chain, an error prone process,
- generally, direct drives are preferred over geared drives, due to the advantages discussed in Section 1.1.3,
- the gimbal structure must be stiff to avoid bending and low-bandwidth resonance.

1.2 Literature Overview

In this section, we provide a review on the current literature about the theme, focusing on novel, relevant works in ISP modeling and LOS control. Since the literature in the field is scattered among many knowledge areas, we propose a separation between *classic* and *modern* techniques for modeling and control.

1.2.1 Classic Techniques in ISP Modeling and LOS Control

Several works have been done in ISP modeling and control. The topic has been extensively studied, mainly driven by military needs in the last century. Despite of possible secrecy issues due to the military nature of this research theme, many relevant papers were written, specially from 1970s through 1990s, and it is still a topic of profound relevance in today's engineering community. In the work of [4], some first concepts about LOS stabilization for mobile VSAT (Very Small Aperture Terminal) antennas are introduced, such as the idea of mass-stabilization with a system of gimbals. According to the author, the ideal mass-stabilized system requires no external torque for *stabilization*. Here, a clear distinction between *stabilization* and *tracking* (or *pointing*) is made. Stabilization stands for coinciding the aimpoint with the LOS with the smallest angular error and vibration (jitter) as possible, while tracking stands for correctly pointing the aimpoint to the LOS, even in the presence of target or base motion. It introduces the idea of a simple control strategy for stabilization, which consists in mounting gyroscopic sensors alongside with the antenna, with their sensitive axes orthogonal to the antenna's aimpoint and applying a control law for annulling the measurement of the gyros. The advantages of this strategy are twofold: (i) the direct stabilization configuration provides direct high-bandwidth measurements of the LOS disturbances and (ii) the gyros operate around their null point, making their scale-factor accuracy a non-critical issue. Steering stabilization systems are also discussed. Because of the particular application, optimization strategies could be used in the design of the tracking or pointing loop, which aims to maximize the power of the signal received by the antenna. Due to the beam power profile around the satellite LOS (shown in Fig. 1.10), real-time extremal seeking optimization could be used as a tracking strategy, for example.

In [2], the principles of ISP-based stabilization for optical imaging systems are presented. In this type of application, the ISP must not only hold the stabilized mass in the desired orientation with minimum angular deviation (jitter), but must also be able to follow the target accurately. Therefore, the *tracking error* performance is another important requirement for this type of application. It addresses the problems of target motion and operational environment: if image processing

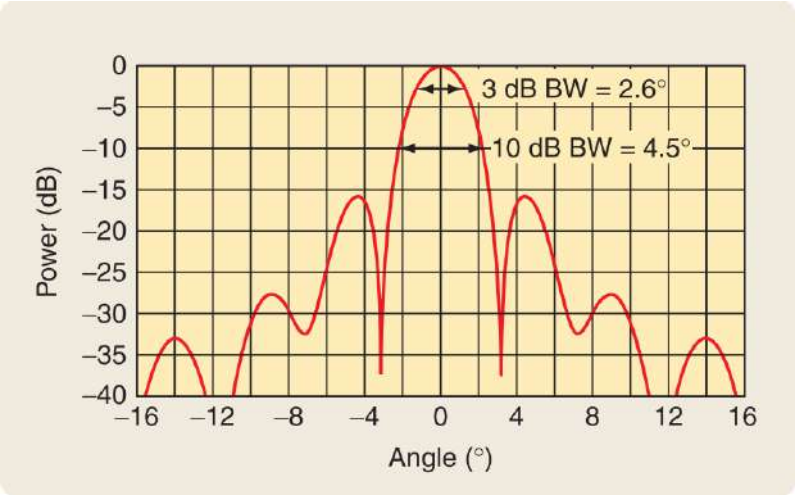


Figure 1.10: Beam profile around the satellite LOS. Image from [4].

algorithms must be used to track the target in space, factors such as computer sample time, camera frame-rate/resolution, and target occlusion can significantly affect the tracking performance. It also introduces the idea of modeling the ISP as an uncoupled double integrator affected by input torque disturbances:

$$I_i \, \ddot{q}_i = \tau_{q_i} + \tau_{d_i} \, , \quad i = 1, ..., n \, , \tag{1.1}$$

where n is the number of DoFs of the ISP mechanism, $\ddot{q}_i \in \mathbb{R}$ represents the i-th gimbal axis acceleration and $I_i \in \mathbb{R}$ is the corresponding axis inertia, while τ_{q_i} and τ_{d_i} are the joint torque and the total disturbance torque, respectively. The disturbance torque τ_{d_i} accounts for all input disturbances, such as mechanical unbalance, friction, mechanical coupling (in geared-drive systems), structural flexibility, and even environmental disturbances such as wind stream and temperature, which influences friction (indirectly creating an additional torque disturbance). Figure 1.11 illustrates some of the main contributions for the the torque disturbances.

The topology discussed for control is a cascade strategy, based in a high-bandwidth inner control loop for the LOS rate, followed by an outer, low-bandwidth loop for the LOS angles (Fig. 1.12). The inner loop tries to compensate the torque disturbances and track the angular rate reference as accurately as possible, while the outer loop keeps the sensor aimpoint in the correct LOS. Typical choices for the LOS rate loop are Proportional (P), Proportional-Integral (PI) or Proportional-Integral-Derivative (PID) controllers, with crescent disturbance rejection performance. Similar choices of controllers may be made for the LOS angle loop, but additional care regarding the stability of the loop must be taken, due to large time delays generally introduced by the tracking algorithms providing the LOS reference signal.

A survey about the ISP technology can be found in [1], from where many of

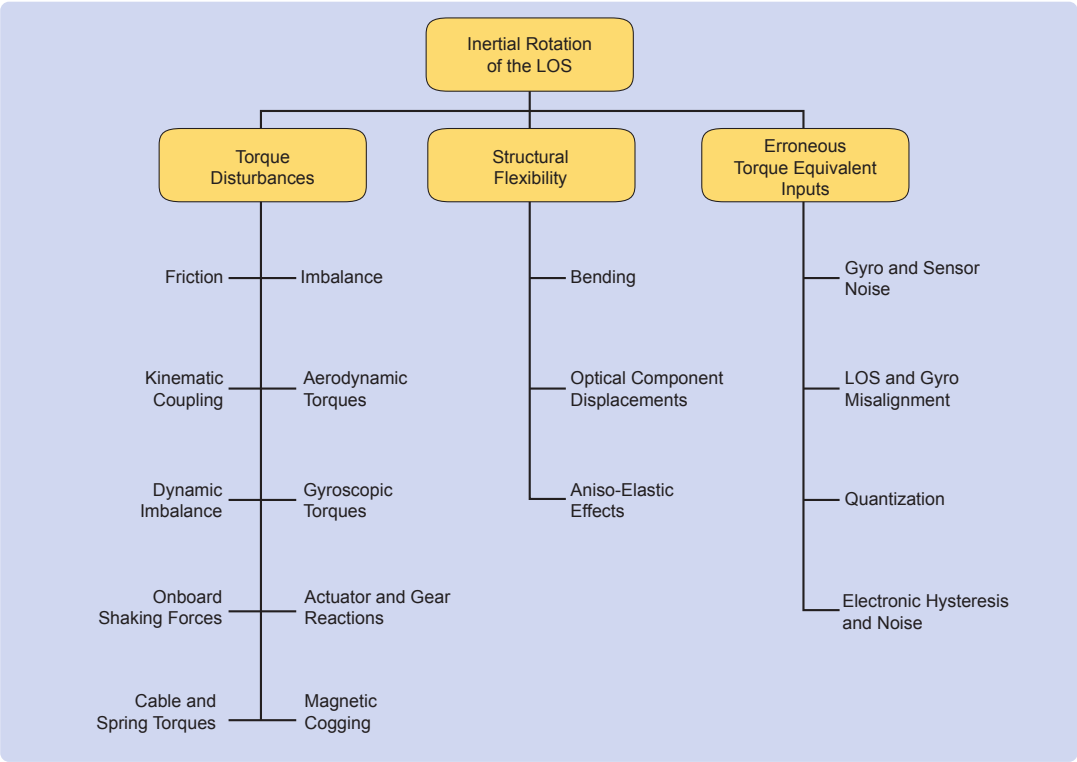


Figure 1.11: Sources of errors in mass-stabilized ISPs, according to [1].

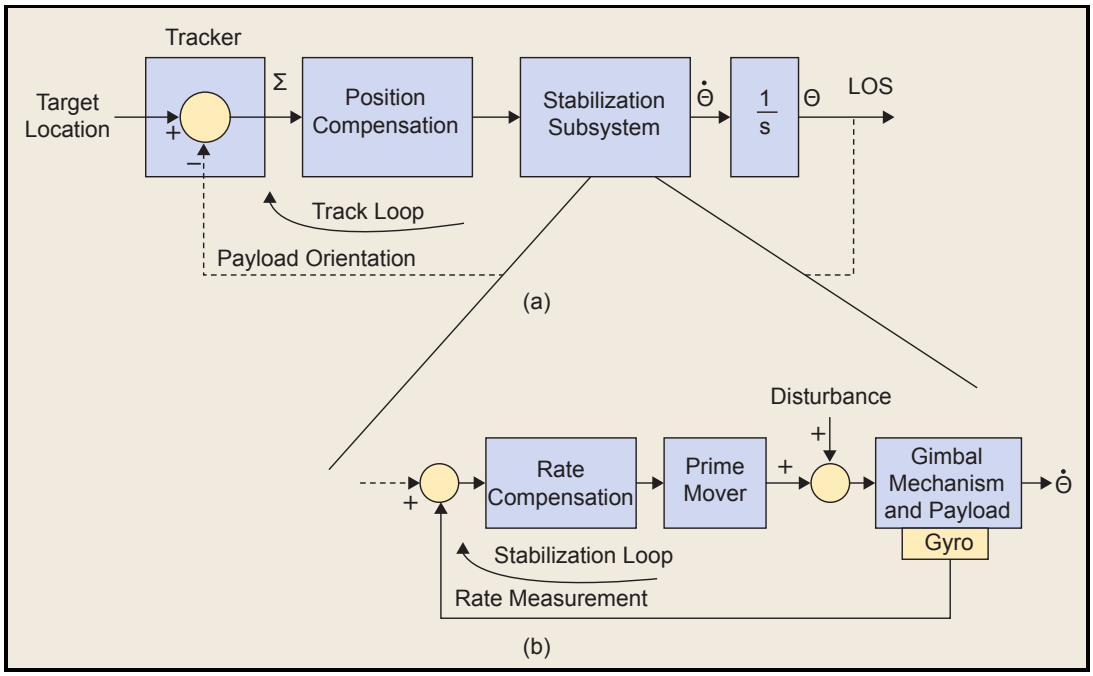


Figure 1.12: Typical LOS control architecture: (a) tracking and (b) stabilization loops. Image from [2].

the concepts and guidelines of Section 1.1 were drawn. Although it discusses the unconstrained dynamics of the multibody systems as an approach for the modeling of the ISP dynamic equations, it does not focuses on the theoretical aspects of the models, providing a wide practical view of the main design instead. It also discusses

the rotational kinematics of the gimbals using the formalism of the Euler angles, based in *elementary* rotation matrices. Although it does not discusses explicitly LOS control algorithms, it gives general guidelines for the design of controllers, based in the type of stabilization (direct or indirect) chosen.

There three works are basically surveys about LOS stabilization and its applications at the time. Nowadays, some new works have been proposed, many still relying on relatively simple models for the description of inertial stabilization systems.

For instance, [17] addresses only the kinematic modeling of ISPs, developing a controller based on the measurements of a 2-DoF gyro installed in direct configuration. The controller basically uses actual gyro information for the computation of the correction angles for yaw and pitch, using the Rodrigues formula from rotational kinematics. The computed angles simply provide the references for two independent Proportional-Proportional Integral (P-PI) controllers for the ISP electric motors, which are modeled as first-order linear systems. This control strategy is a simplification over the common cascade structure, since the inner loop is simply designed to track the desired motor angles.

These works rely on the simple dynamic model of (1.1) for the design of stabilization LOS control loops. More recent works, however, study more complex mechanical phenomena that occur due to structural imperfections in the gimbals, such as the *cross-coupling* effect (the dynamic effect on one axis by the rotation of another) that arises from the presence of dynamic unbalance. In [5], the effects of mobile base motion with and without gimbal dynamic unbalance are analyzed for a 2-DoF ISP. The article uses the Newton-Euler's equations of motion to derive explicit formulas for the disturbance torques due to the kinematic coupling between the moving base and the payload. Figure 1.13 illustrates how the disturbance torques affect both control stabilization loops, and depend explicitly on the motion of the base.

Even using a simple argument, it is easy to show how the *accelerations* of the moving base play an important role on the ISP dynamics. If the base rotational amplitude and angular velocities are small, one can approximate the kinematic relation between the joint motion, the vehicle motion and the payload motion by:

$$\dot{\eta}_{c_2} \approx \dot{q} + \omega_{0b}^{b}\,, \tag{1.2}$$

where η_{c_2} is some minimum parametrization for the camera orientation and ω_{0b}^{b} is the vehicle angular velocity. With this simplification, (1.1) could be rewritten (in

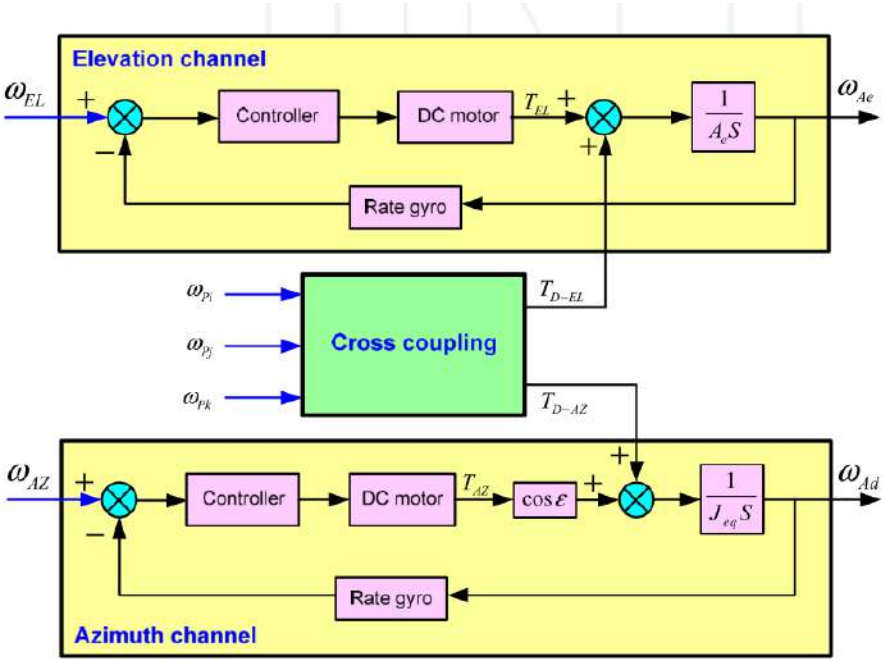


Figure 1.13: The elevation and azimuth stabilization loops for a 2-DoF ISP, showing the cross coupling effect due to mobile base motion generating disturbance torques to the elevation (pitch) and azimuth (yaw) channels. Image from [5].

vector form) in terms of the *operational space dynamics* as:

$$I\ddot{\eta}_{c_2} = \tau_q + \underbrace{\tau_d + I\,\dot{\omega}^b_{0b}}_{disturbance}\,. \tag{1.3}$$

Notice that the vehicle angular acceleration would be considered as part of the disturbance vector in the payload orientation dynamics.

In high accuracy applications, the unmodeled effects composed in disturbance torque vector can represent significant contributions, and simple linear controllers may not suffice. That is why an increasingly number of works have been done in an attempt to provide more realistic descriptions of the ISP dynamics. After all, an accurate dynamic model can provide a better understanding of the system and allow the implementation of more sophisticated control strategies. From the modeling perspective, an ISP can be considered as a robot manipulator with $n = 2$ revolute joints ($n = 3$, in the case of roll motion) forming a gimbal structure and installed in a non-inertial base. Therefore, an adequate mathematical framework for this type of system is the Vehicle-Manipulator System (VMS) framework, developed in [18, 19]. It consists basically in applying the Euler-Lagrange equations to systems of interconnected rigid bodies. [19] focuses on the dynamic equations of robot manipulators installed on a vehicle, such as vessel or ship. These equations

can be written compactly as:

$$M_{qq}\,\ddot{q} + C_{qq}\,\dot{q} + G_q + \underbrace{M_{qV}\,\dot{V}_0 + C_{qV}\,V_0}_{base\,motion\,terms} + \tau_f = \tau_q\,, \tag{1.4}$$

where $V_0, \dot{V}_0 \in \mathbb{R}^6$ are the velocity and acceleration twists of some fixed point in the vehicle, $q, \dot{q}, \ddot{q} \in \mathbb{R}^n$ are the vectors of robot joint angles, velocities and accelerations, and $\tau_q \in \mathbb{R}^n$ is the vector of joint torques. The term τ_f accounts for the general torque contribution of friction in the gimbal dynamics. From standard texts in robotics literature such as [20, 21], (1.4) can be recognized as the dynamic equation for robots, generalized for accounting the motion of the base. Also note that the last four terms on the left-hand side of (1.4) represent a torque disturbance due to the nonlinearities of the system *and* the motion of the vehicle. However, although the complete model (1.4) may be quite complex, reasonable approximations can be made. For example, under the assumption that the ISP is perfectly balanced, $C_{qq} \approx 0$, $G_q \approx 0$, and the system resembles a simple double integrator with a torque input and a disturbance due to the base motion.

It is worth to say that, although (1.4) represents the system dynamics in a simple matrix differential equation, the internal expressions for the matrices can grow quite complex depending on the number of simplifying assumptions made. However, efficient algorithms can be used to compute the model matrices numerically. The advantage of this approach is that all nonlinear phenomena associated to systems of interconnected rigid-bodies (such as axes cross-coupling and dynamic unbalance introduced in [5]) are naturally represented here in this compact form. Even friction models can be easily introduced as an additional negative joint torque disturbance.

1.2.2 Modern Techniques in ISP Modeling and LOS Control

Other control methods exist besides the decentralized control in inner stabilization and outer tracking loops. In some applications, high accuracy/precision must be achieved and/or the magnitude of the unmodeled dynamics is too high to be compensated by high gain attenuation. Other limitations may be present as well, such as the existence of *representation singularities* in applications where the mobile vehicle where the ISP is installed requires a wide range of motion, or specific problems arising from image stabilization applications. Therefore, more involved control techniques are still being studied for the LOS control problem.

In [22], an Image-Based Visual Servoing (IBVS) scheme is proposed as an approach to control the LOS of an airborne double-gimbal ISP for image tracking applications, as shown in Fig. 1.14. The authors provide an analysis of a cascade

controller composed of an inner stabilization loop and an outer control loop that tracks a selected target in the image, and demonstrate the robustness of the strategy to high computation delays due to the image processing algorithms.

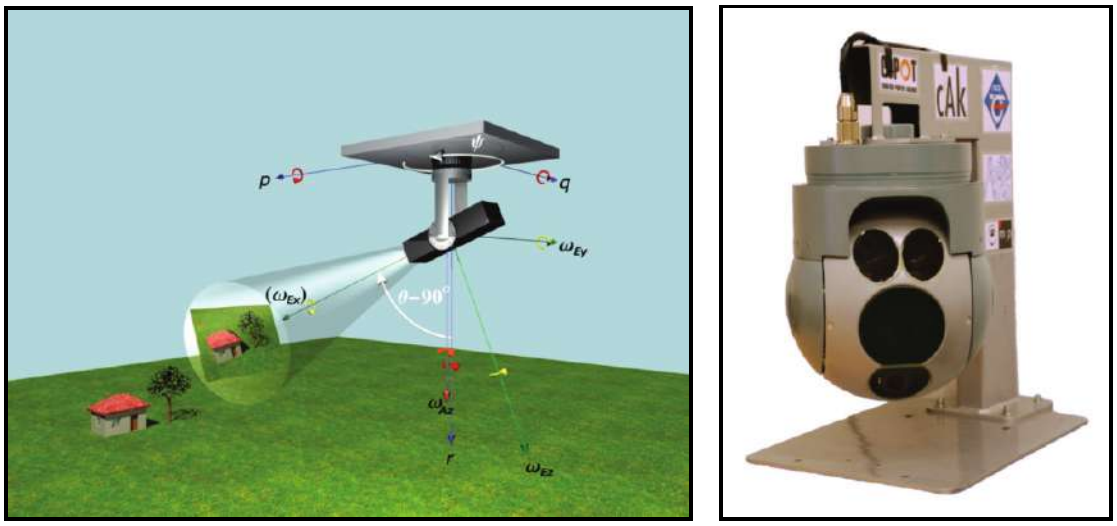


Figure 1.14: Tracking of a target at the image using a 2-DoF ISP. Images from [6].

It also introduces the idea that the ISP complete kinematic model is composed of a combination of its own rotational motion states and an image disturbance caused by the *linear velocity* of the mobile base, which cannot be directly compensated due to the lack of sufficient DoFs in the ISP mechanism. Besides, it criticizes a common decoupled controller used in literature and commercial products for the outer tracking loop: basically, it commands only the elevation loop to track the vertical dimension, and the azimuth loop to track the horizontal dimension. However, the authors claim that this is a naive approach due to the natural coupling that exists between the image axes and the gimbals due to the perspective projection model. With high-delay visual tracking algorithms, this technique could lead to a decrease in performance, since the existing coupling is not properly compensated by the controller.

Therefore, the authors propose a coupled tracking controller that uses a direct-placed inertial sensor and image-space information of the target to compensate the nonlinearities introduced in the system by the camera model, effectively decoupling the vertical and horizontal dimension dynamics from each other. It also compensates the translational motion of the vehicle by feeding back to the controller the linear velocity of the camera. The proposed controller achieves asymptotic convergence of the image-space errors, supposing perfect knowledge of the system model. This control strategy consists of computing the ideal angular velocity references to the yaw and pitch motors using the pseudo-inverse of the system *image Jacobian* and the image errors, a common strategy in IBVS control. Experiments made on the real

system of Fig. 1.14 show a slightly better dynamic response than the one obtained with the decoupled controller. It is worth noticing that the *coupling* that arises in the controller is due to the introduction of image-space coordinates, and not due to mechanical coupling between the yaw and pitch axes. That is, dynamically, the model is still given by (1.1).

The same authors propose a feedforward controller for compensating the *acceleration* of the mobile base in [23]. The 2-DoF gimbal mechanical model is the standard decoupled dynamic model of (1.1), but the electrical dynamics of the motors were also considered. The authors claim that *friction* indeed plays a major role in the dynamic response of the platform, and use the LuGre model [24] for simulating the friction response of the system. For the design of the feedforward controller, an accelerometer was placed in the mobile base, and its measurements were propagated through the gimbal structure, computing the acceleration of the payload. The gimbal parameters were identified using least-squares techniques based on the dynamic response of the gimbal angles to a given voltage command. This work suggests the *adaptation* of the kinematic parameters of the structure as an alternative for the limited accuracy of the parameter identification techniques, which could result in better acceleration rejection performance. In [25], these techniques were summarized in a doctoral thesis, covering inertial stabilization, ISP parameter estimation and visual servoing control schemes for aerial surveillance applications.

Regarding the search for accurate mechanical models, a common approach from robotics is *computed torque* control, which consists of canceling all the nonlinear terms in (1.4). If a complete model of the ISP mechanism as in (1.4) is available, the stabilization loop can be designed as a torque control law of the type:

$$\tau_q(t) = \underbrace{M_{qq}\, v(t)}_{feedback} + \underbrace{C_{qq}\,\dot{q} + G_q + M_{qV}\,\dot{V}_0 + C_{qV}\,V_0 + \tau_f}_{feedforward} \,. \tag{1.5}$$

Clearly, substituting (1.5) into (1.4) results in a double integrator with the signal $v(t)$ as an input. If $v(t)$ is chosen to be a PD control in the joint space, this strategy guarantees asymptotic stability in the joint space errors, *supposing* the perfect knowledge of all dynamic parameters.

This technique is proposed in a recent work [26], where an orientation control scheme for a 3-DoF gimbaled platform is used for stabilization of film and broadcast cameras. In this paper, the ISP dynamics is obtained by the Euler-Lagrange formalism for multibody systems in a similar way than in [27], which is studied in deeper detail in [19]. The *quaternion* formalism was used to represent body orientation, which prevents the problems of representation singularities arising from the use of *minimum* representations, such as some set of Euler angles. The result-

ing dynamic equation for the system is completely equivalent to (1.5), with base disturbances represented as the time-derivatives of the base pose represented in $\mathbb{R}^7$ constrained space, composing 3D position and 4D orientation in quaternion space. This reaffirms the obvious equivalence between the ISP model and models of robots installed in non-inertial platforms. Joint friction torque is also taken into account by a simple static model composing viscous friction and a continuous approximation for Coulomb friction using the hyperbolic tangent function. A strong stabilization technique was proposed, using two inertial sensors (one at the vehicle and other at the payload). By assuming complete knowledge of the ISP kinematics, the authors propose a fusion sensor technique using a modified Kalman filter to combine the direct and indirect measurements obtained by the two sensors, achieving a higher precision estimation for the camera orientation [28]. The control strategy employed consists in the sum of two control terms: (i) a *feed-forward* term for the cancellation of some of the base disturbance terms and system non-linearities (base linear acceleration, gravity and estimated friction); and (ii) a *feedback* PD term for stabilizing the error equation in the *joint space*. Therefore, an additional step for computing the the *joint space errors* from the *quaternion space errors* (given by the desired LOS angles and velocities) in real time is needed. Both errors are related by means of the pseudo-inverse of a Jacobian matrix. However, this is not an exact relation, since the simplifying assumption of small joint space errors was made. Small norm terms such as Coriolis torques were neglected, and the authors justify the obtained boundedness of the error signals by invoking *robustness* properties of the computed torque control for fixed robot manipulators [29] and *boundedness* of all uncanceled terms. Simulations and experiments using a prototype ISP where presented.

The computed torque technique relies in a troublesome assumption. To achieve exact cancellation, all parameter-dependent terms used in (1.5) (the model *dynamics parameters*) must be exactly known. Otherwise, torque disturbances due to imperfect cancellation will occur. Therefore, the performance obtained by using this strategy may not be significantly better than that obtained by a simple model with linear controllers presented in Section 1.2.1 if a high-accuracy parameter identification technique is not used. Fortunately, there are many available techniques to deal with the problem of model parametric uncertainty. *Adaptive control* is one of them, and was widely studied in the robotics literature during the 80's. It consists in designing a time-varying, model-based controller that ensures stability and convergence of the tracking errors even in the presence of parametric uncertainty, by applying some method for the online estimation of the controller parameters (usually based in some prediction error metric).

The problem of adaptive attitude control invokes back to the seminal work [30]

by John T. Wen in the 90's. It discusses the theoretical problem of dynamic attitude control for a single rigid-body using the unit quaternion representation. In this work, the problem was tackled by using the same structure as in [26]: a quaternion/angular velocity PD feedback term plus a feed-forward compensation for the system dynamic nonlinearities. The global asymptotic stability of an adaptive scheme accounting for uncertainty in the body mass matrix is also proposed, using quaternion and angular velocity errors. Although it does not address the problem of attitude control for multibody systems, it was the first work to address the problem of *adaptive control* of rigid-body orientation using the unit quaternion formalism. A latter work by F. Lizarralde and John T. Wen [31] proposes an attitude control scheme using the unit quaternion representation without the need for angular velocity measurements. The passivity properties of the closed-loop system are explored in order to prove the global asymptotic stability of a control law that is based only in the quaternion error and on a SPR linear filter. The results are also applied for the task-space control of a non-redundant robot manipulator on a *fixed* base using a control law based on a PD-like plus a gravity compensation term, where only the end-effector pose and the robot joint angles where measured. However, the problem of parameter uncertainty is not tackled here.

Recently, Cabarbaye [32] developed an alternate adaptive control for an inertially stabilized payload with unknown inertia matrix, using the unit quaternion formalism and quaternion/angular velocity feedback errors. Since the payload is mounted on an ISP, the control method consists in projecting the computed torque control signals into the ISP axes and using them as reference signals to the ISP motor drivers. Therefore, it does not take into account the dynamic disturbances caused by the ISP dynamics itself (only of the payload), as in [26]. The work makes many assumptions about the ISP construction that may be unrealistic, depending on the system: (i) mechanically balanced system; (ii) symmetrically distributed mass and (iii) negligible joint friction. Furthermore, the stability analysis is incomplete and misleading, and only simulation results are available. As far as I am concerned, no other works tackled the mass-stabilization LOS control problem with unknown or uncertain system dynamic parameters using adaptive controllers. Also, it appears to have no works on the adaptive control of uncertain robot manipulators in a moving platform as well.

Other recent works use different kinds of modern techniques to tackle the LOS rate stabilization control problem. For example, [7] proposes a self-tunning PID-type fuzzy controller as an alternative to the common PID control used in the ISP stabilization loop for both azimuth and elevation gimbals. The authors use their previous model of a two axes gimbal system developed in [5] to compare the per-

formance between the two control methods. The fuzzy feedback controller consists of four main components: (i) a fuzzyfication interface, (ii) a *knowledge base*, (iii) an inference mechanism and (iv) a defuzzyfication interface, as seem in Fig. 1.15.

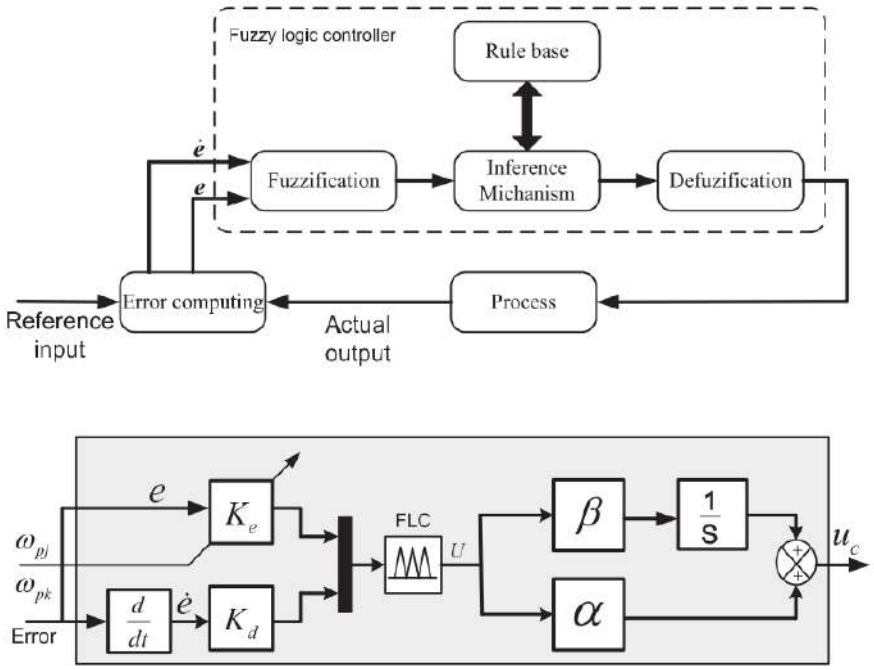


Figure 1.15: General structure for fuzzy control, and actual PID fuzzy controller architecture. Image from [7].

The fuzzyfication interface consists of a set of empirical rules for the selection of the output activation levels based on the input magnitudes, based in some knowledge base. In other words, it typically converts the input data into some suitable linguistic values. A typical approach consists in defining *membership functions* for classifying the *degree of membership* of the input variables into some type in a linguistic set, as seem in Fig. 1.16.

The rule base applied for the level selection of the internal control variable U is given by table (Fig. 1.17) representing the knowledge base used for the problem. It selects the fuzzy value of the output U based on the fuzzy values of the rate error and its approximate derivative, obtained from the membership functions.

The defuzzyfication interface stands for the inverse of the map showed in Fig. 1.16, and is responsible for selecting numerical values for the output U, given its linguistic value computed from the knowledge base in Fig. 1.17. This process computes the actual numeric value for U, while the actual current control law is given by a PI controller with the numeric value of U as its input.

Also, [7] used an empiric adaptation process for the online adjustment of the

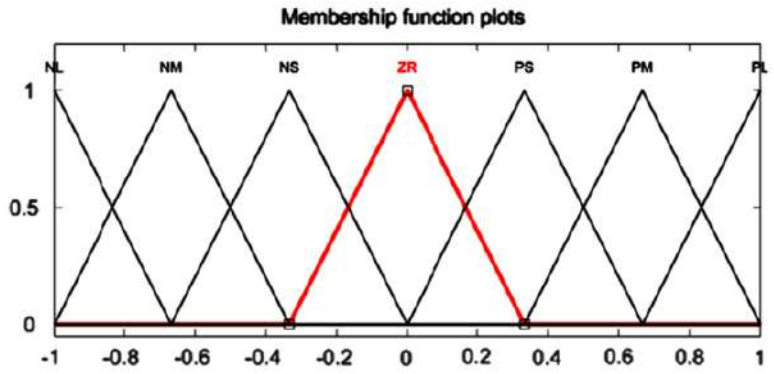


Figure 1.16: Membership functions for arbitrary inputs in the closed interval $[-1, 1]$. They are classified into some element of the set {NL,NM,NS,ZR,PS,PM,PL}, which stands for *negative/positive large, medium, small* and *zero*. Image from [7].

Fuzzy PID type rule base.

$e/\dot{e}$	**NL**	**NM**	**NS**	**ZR**	**PS**	**PM**	**PL**
NL	LN	LN	LN	LN	MN	SN	ZE
NM	LN	LN	LN	MN	SN	ZE	SP
NS	LN	LN	MN	SN	ZE	SP	MP
ZR	LN	MN	SN	ZE	SP	MP	LP
PS	MN	SN	ZE	SP	MP	LP	LP
PM	SN	ZE	SP	MP	LP	LP	LP
PL	ZE	SP	MP	LP	LP	LP	LP

Figure 1.17: Rule base for the level selection of the internal control variable U. Image from [7].

proportional and integral gains of the fuzzy PID controllers, based on the yaw and pitch angular velocity components of the base motion. Simulation results appear to demonstrate the higher performance of the this method in terms of rate overshoot and settling time, when compared to simple PID controllers.

Other *variable structure* approach for LOS control was proposed by [33], which consists of a *sliding mode* control strategy for the stabilization loop of a 2-DoF ISP with unknown disturbances. The ISP system is modeled in a similar way than in [7], but the resulting space-state equations comprise the pitch/yaw rates and motor currents as its first and second state, respectively. Since the control uses only one rate sensor in a direct configuration, and no yaw and pitch encoders or Hall sensors, the resulting dynamic equations depend on unmeasured variables (the ISP joint velocities and the motor currents), as long as all other dynamic disturbances, such as the base motion coupled terms and gimbal friction. Therefore, both dynamic equations have disturbances that sum to their respective state equations. Since the control inputs are the motor driver voltages, it only affects the second state equation

(the motor current dynamics). In the context of *robust* control, a matched disturbance is a disturbance that appears in the same dynamic equation as the control input. Therefore, the whole system is affected by matched and unmatched disturbances. The authors propose two high-order sliding mode observers (HOSMO), one for the estimation of the second unmeasured state (the motor current) and its time derivative, and another for the estimation of a composite disturbance, consisting in a combination of the matched and unmatched system disturbances. Next, the proposed controller for the motor driver voltage is based on the Non-singular Terminal Sliding Mode (NTSM) control design [34], and uses an estimated NTSM-type sliding surface constructed from the measured inner gimbal attitude and the estimated motor current state. The authors prove by the direct method of Lyapunov that, under the assumption of boundedness of the second derivative of the motor currents and the composite disturbance, the proposed controller guarantees that the angular rates converge to their desired values in *finite* time, which is a desirable characteristic of the sliding mode approach. Computer simulations seem to demonstrate the superior performance of this method when compared to a PI controller.

The works of [7] and [33] only deal with the stabilization control problem. They consider the stabilization of the angular rates as the head part of the LOS control design, and propose that simple control techniques can be used in the outer control loop, for the tracking of the LOS angles. This is not entirely true, as it is evident from [6], which deals directly with complications that may arise in the design of the outer loop due to the image-space transformation. Therefore, the tracking loop design can also be a challenging problem. While the stabilization loop is highly affected by the dynamic disturbances of the platform, the tracking loop can be mainly dependent on the ISP *kinematics*. For example, in indirect configurations, the LOS angles are computed by combining the measured base orientation with the gimbal joint angles to estimate the payload orientation. If the ISP kinematics is uncertain, the deviation to the real LOS angles could be very high, resulting in a persistent bias that cannot be simply eliminated by the controller; instead, a calibration technique must be used to identify the ISP kinematic deviations (such as axes misalignment) with acceptable accuracy.

Another possible solution to the problem of *kinematic uncertainty* lies again on adaptive control. If enough sensors are present in the system, information about the its kinematic structure could be retrieved online, adapting the parameters of the tracking loop for a better tracking performance. Although no works have been done specifically on the adaptive control of ISPs with uncertain kinematics, the adaptive control problem for robot manipulators with uncertain kinematics gave rise to a very active and promising field of study, with important recent works such

as [35, 36]. These works addressed the problem of robot control in the task space with uncertain dynamics and kinematics. Although not immediately applicable to LOS control (which involves the kinematics of the vehicle as well as of the ISP), their approach seen to be directly extensible to robot manipulators in a moving base, and therefore, to ISP systems.

A last remark must be made in the use of complex control strategies. Although tempting at first sight, their use must be studied with care. When complexity arises, generally the difficulty in setting the controller parameters (gains) also arises, and thus the difficulty in setting those parameters in a way that results in an acceptable control performance. Once more, we reaffirm the importance in establishing the performance requirements for the LOS control problem: one of the main reasons why classic control methods where used for so long is probably because they provide enough performance for a wide range of applications.

1.3 Objectives of this Work

As it can be seen, in high precision/accuracy applications, complex control strategies may be necessary. Model-based control such as well-known feedback-linearization techniques can be used if an accurate dynamic model of the system is available. If this is not the case, robust control techniques must be employed. This seems to be the modern approach that most researchers are taking to deal with the problem of LOS control.

Since LOS control can be reduced to an orientation control problem, a critical issue in its formulation regards the appropriate representation for the camera orientation in space. In applications where the amplitude of the angular motion of the vehicle is small, such as in wheeled vehicles, ships and marine vessels, *minimum* representations for the camera orientation such as RPY or Euler angles may suffice, because the small amplitude of the base motion generally avoids the camera orientation to get near singular configurations.

However, in applications with large angular motion amplitudes, such as in UAVs, quadrirotors and satellites, it is very difficult to avoid representation singularities when using minimal representations for orientation. The unit quaternion formalism is a non-minimal angular representation for rigid-body rotations that provides an adequate framework for describing the kinematic model for rotation.

The general objectives of this work are:

1. The development of an appropriate mathematical model for an ISP mounted on a mobile vehicle,

2. The development of model-based control strategies for the inertial stabilization of a sensor and the LOS tracking of a moving target considering any kind of vehicle,

3. To evaluate the performance of the proposed strategies with respect to parametric uncertainties on the ISP model,

4. To provide realistic simulation results for the proposed controllers, and to compare their performance with respect to commonly used techniques.

Particularly, feedback linearization controllers are proposed for both indirect and direct configurations of the ISP, and their robustness against parametric uncertainty is studied and validated. Additionally, two sliding mode controllers based on the super-twisting algorithm are proposed as a powerful alternative to the feedback linerization controllers, providing *ideal* disturbance rejection and robustness properties against parametric uncertainties. Besides, since the proposed solutions are based on the quaternion formalism, they do not suffer from representation singularities.

Chapter 2

Modeling of Inertial Stabilization Platforms

In this chapter, we introduce a modeling framework based on rigid-body dynamics, allowing an accurate mechanical model for the ISP the dynamic disturbances acting on its base. We cover the mechanism and LOS kinematics in the first section, and then describe in detail the theory of rigid-body dynamics, used to derive the ISP equations of motion.

2.1 Definitions

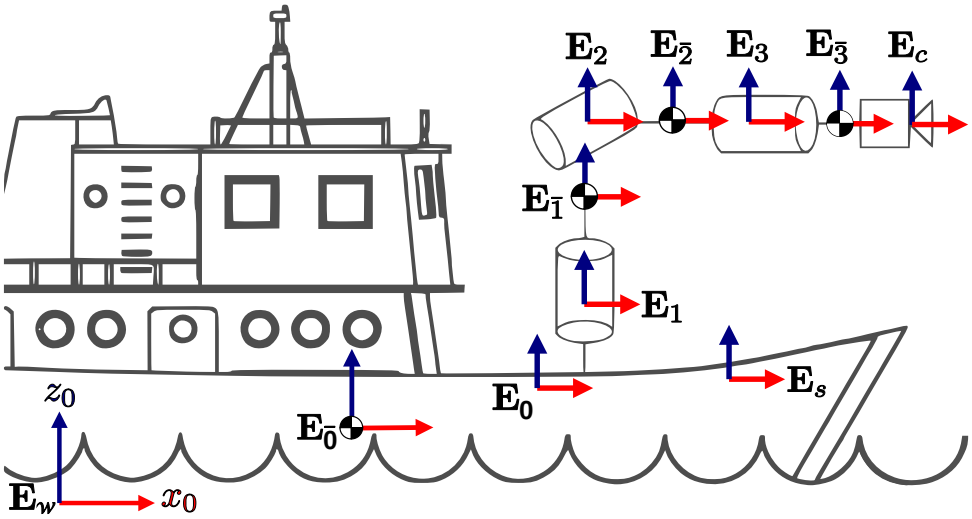


Figure 2.1: Frame conventions for a 3-DOF ISP installed on a vessel.

In the following, we introduce the definition of the main variables that will be used to derive the dynamic and kinematic models used in this chapter. First, we need to define some useful reference frames, illustrated in Fig. 2.1.

- $\mathbf{E}_w$: world (inertial) frame, arbitrarily located;
- $\mathbf{E}_0$: vehicle base frame, arbitrarily located on the vehicle;
- $\mathbf{E}_{\bar{0}}$: vehicle frame located on its Center of Gravity (CG);
- $\mathbf{E}_i$: frame attached to link i ($i = 1, 2, 3$) located on joint j axis;
- $\mathbf{E}_{\bar{i}}$: frame attached to i ($i = 1, 2, 3$) located on the link CG;
- $\mathbf{E}_c$: frame located in the camera's optical focus, attached to the last link;
- $\mathbf{E}_t$: target frame, located on an arbitrary position;
- $\mathbf{E}_s$: inertial sensor frame. Can be attached to the camera or to the vehicle.

The dynamic formulation will be derived according to the base frame $\mathbf{E}_0$, attached to the vehicle. If the INS is installed on the vehicle (indirect stabilization method), then the base frame is equivalent to the INS frame ($\mathbf{E}_0 \equiv \mathbf{E}_s$). In the case of direct stabilization, the base frame is equivalent to the first joint frame ($\mathbf{E}_0 \equiv \mathbf{E}_1$), and the IMU frame $\mathbf{E}_s$ is fixed to the last link of the ISP, in an arbitrary position.

In order to express relations of motion (pose, velocity and acceleration) between one frame relative to another, subscript indexes are used to represent the origin and destination of the vector. Superscript symbols denote in which coordinate system this vector is represented. For example, $\begin{bmatrix} x_0 & y_0 & z_0 \end{bmatrix}^\mathsf{T}$ is the position of the vehicle frame $\mathbf{E}_0$ relative to the inertial frame $\mathbf{E}_w$ represented in the inertial frame coordinates, while V_0^0 is the vehicle velocity twist (linear and angular velocities) relative to the inertial frame represented in the local coordinates.

Define the infinite set $\bar{\mathbb{N}}_0 = \{\bar{0}, \bar{1}, \bar{2}, \ldots\}$. In the following, unless otherwise stated, the indexes $i, j, k \in \mathbb{N}_0 \cup \bar{\mathbb{N}}_0 \cup \{c, s, t\}$, where $\mathbb{N}_0$ is the set of natural numbers. Define the following matrices, vectors and scalars:

- $\mathbf{x}_i, \mathbf{y}_i, \mathbf{z}_i \in \mathbb{R}^3$ are the canonical unit vectors of $\mathbf{E}_i$;
- $R_{ij} \in SO(3)$: rotation matrix describing the orientation of $\mathbf{E}_j$ relative to $\mathbf{E}_i$;
- $\mathbf{S}(v) \in so(3)$: cross-product operator acting on a 3D vector $v \in \mathbb{R}^3$;
- $\mathsf{se}(v) \in se(3)$: twist operator acting on a 6D vector $v \in \mathbb{R}^6$ of twist coordinates.
- $\lfloor v \rceil^\alpha$: $\mathbb{R}^n \to \mathbb{R}^n$, where its elements are given by $|v_i|^\alpha \, sgn(v_i)$, with $v_i \in \mathbb{R}$ being the elements of $v \in \mathbb{R}^n$ and $\alpha \in \mathbb{R}$;
- $p_{ij}^k \in \mathbb{R}^3$: position vector from the origin of $\mathbf{E}_i$ to the origin of $\mathbf{E}_j$, written in $\mathbf{E}_k$;

- $v_{ij}^k \in \mathbb{R}^3$: linear velocity vector from $\mathbf{E}_i$ to $\mathbf{E}_j$, written in $\mathbf{E}_k$;
- $\omega_{ij}^k \in \mathbb{R}^3$: angular velocity vector from $\mathbf{E}_i$ to $\mathbf{E}_j$, written in $\mathbf{E}_k$;
- $V_{ij}^k \in \mathbb{R}^6$: velocity twist (linear and angular velocities) from $\mathbf{E}_i$ to $\mathbf{E}_j$, written in $\mathbf{E}_k$;
- $g_{ij} \in SE(3)$: homogeneous transformation matrix from frame $\mathbf{E}_j$ to frame $\mathbf{E}_i$;
- $h_i^j \in \mathbb{R}^3$: unit vector defining the rotation axis of joint i, represented in $\mathbf{E}_j$;
- $m_i \in \mathbb{R}$: mass of body i ($i \in \mathbb{N}_0$);
- $I_i^j \in \mathbb{R}^{3\times 3}$: inertia tensor of body i represented in $\mathbf{E}_j$ ($i, j \in \mathbb{N}_0$);
- $\bar{I}_i^j = \begin{bmatrix} m_i\,\mathsf{I}_3 & -m_i\,\mathbf{S}(p_{ji}^j) \\ m_i\,\mathbf{S}(p_{ji}^j) & I_i^j \end{bmatrix}$: generalized inertia matrix of body i represented in $\mathbf{E}_j$ ($i, j \in \mathbb{N}_0$);

Recall that the inertia tensors and the generalized inertia matrices are constant if represented on local frames. Without loosing generality, instead of choosing the CG frames aligned with the principal axes of inertia (where the inertia tensor is diagonal), we choose frames $\mathbf{E}_{\bar{i}}$ and $\mathbf{E}_i$ to be aligned ($i \in \mathbb{N}_0$), for simplicity. Therefore, off-diagonal terms have to be introduced the inertia tensor:

$$I_i^{\bar{i}} = \begin{bmatrix} (I_{xx})_i & (I_{xy})_i & (I_{xz})_i \\ (I_{xy})_i & (I_{yy})_i & (I_{yz})_i \\ (I_{xz})_i & (I_{yz})_i & (I_{zz})_i \end{bmatrix} .$$

If the inertia tensor $I_i^{\bar{i}}$ of any rigid-body represented on its CG is known, it is easy to calculate the inertia tensor represented on any $\mathbf{E}_j$ using Steiner's theorem [37]:

$$I_i^j = I_i^{\bar{i}} - m_i \left(p_{j,\bar{i}}^j \, p_{j,\bar{i}}^{j\,\mathsf{T}} - p_{j,\bar{i}}^{j\,\mathsf{T}} \, p_{j,\bar{i}}^j \, \mathsf{I}_3 \right) , \tag{2.1}$$

given that $\mathbf{E}_{\bar{i}}$ and $\mathbf{E}_j$ are aligned and located on the same rigid body. Nevertheless, the vehicle generalized inertia matrix does not appear in the manipulator dynamic equations, and thus does not need to be known for the ISP control.

Note that $I_i^{\bar{i}}$ is a constant matrix since it is fixed to rigid body i. Define $\mathbf{E}_{\bar{i}}^*$ as another frame located on the body CG, but aligned with the principal axis of inertia of the body. With respect to this frame, the inertia matrix is not only constant but also *diagonal*, and is denoted by $I_i^{\bar{i}*}$. Its diagonal terms are known as the *main components of inertia* of the body. If R denotes an arbitrary rotation

matrix from $\mathbf{E}_{\bar{i}}^{*}$ to $\mathbf{E}_{\bar{i}}$, then the relation between $I_i^{\bar{i}*}$ and $I_i^{\bar{i}}$ is given by the similarity transformation:

$$I_i^{\bar{i}} = R\, I_i^{\bar{i}*}\, R^{\mathsf{T}}\,.$$

The representation of vectors in the inertial frame $\mathbf{E}_i$ will herein have the superscript index omitted for a more compactness. Next, we define the following linear and angular positions:

- $q = \begin{bmatrix} q_1 & \cdots & q_n \end{bmatrix}^{\mathsf{T}}$ is a $\mathbb{R}^n$ vector representing the n joint angles.
- $\eta_i = \begin{bmatrix} \eta_{i_1}^{\mathsf{T}} & \eta_{i_2}^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}} = \begin{bmatrix} p_i^{\mathsf{T}} & \eta_{i_2}^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}} = \begin{bmatrix} x_i & y_i & z_i & \phi_i & \theta_i & \psi_i \end{bmatrix}^{\mathsf{T}}$ is the $\mathbb{R}^6$ pose vector of $\mathbf{E}_i$ relative to the inertial frame represented in inertial coordinates, where $\eta_{0_1} \in \mathbb{R}^3$ is the inertial position and $\eta_{0_2} \in \mathbb{R}^3$ is some *minimal* parametrization for orientation.
- $\xi_i = \begin{bmatrix} \eta_i^{\mathsf{T}} & q^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}}$ are the $\mathbb{R}^{6+n}$ generalized coordinates of the vehicle-manipulator system. [1] associated to frame $\mathbf{E}_i$, where $\eta_i \in \mathbb{R}^6$ stands for the frame inertial pose in *minimal* coordinates and $q \in \mathbb{R}^n$ stands for the manipulator (ISP) joint angles.

Besides the position states, it is also needed to define velocity states:

- $\dot{q} = \begin{bmatrix} \dot{q}_1 & \cdots & \dot{q}_n \end{bmatrix}^{\mathsf{T}}$ is a $\mathbb{R}^n$ vector representing the n joint angular velocities.
- $V_i^i = \begin{bmatrix} v_i^{i\mathsf{T}} & \omega_i^{i\mathsf{T}} \end{bmatrix}^{\mathsf{T}}$ is the body velocity twist of frame $\mathbf{E}_i$, composed of its body linear and angular velocities ($v_i^i \in \mathbb{R}^3$ and $\omega_i^i \in \mathbb{R}^3$) relative to the inertial frame.
- $\zeta_i = \begin{bmatrix} V_i^{i\mathsf{T}} & \dot{q}^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}} \in \mathbb{R}^{6+n}$ are the quasi-velocities [19] of the vehicle-manipulator system with respect to the frame $\mathbf{E}_i$, composed of the frame body twist V_i^i and the manipulator joint velocities $\dot{q} \in \mathbb{R}^n$.

With these notations and quantities defined, we are able to proceed with the derivation of kinematic and dynamic models for the system.

[1] We will sometimes refer the ISP as a manipulator in this text due to the robotics approach we are using.

2.2 Kinematics of Inertial Stabilization Systems

In this section, we get into the details of the mechanism kinematics, which establishes the relations of the admissible configurations and velocities of the system, without concerning the causes of motion (forces and torques). First, we cover the general aspects of ISP kinematics, such as forward and differential kinematics. Then, the kinematics of the LOS is derived, both in RPY and quaternion form.

2.2.1 Rigid-body Transformations

To uniquely specify the rigid body pose relative to some reference frame (eg. inertial space), one needs to know its position and orientation relative to this frame. This can be characterized by a homogeneous transformation g_{ij} containing a translation vector p_{ij}^i and a rotation matrix R_{ij} [21]:

$$g_{ij} = \begin{bmatrix} R_{ij} & p_{ij}^i \\ 0 & 1 \end{bmatrix}, \quad p_{ij}^i = \begin{bmatrix} x_{ij}^i \\ y_{ij}^i \\ z_{ij}^i \end{bmatrix} \in \mathbb{R}^3, \quad R_{ij} \in SO(3), \quad g_{ij} \in SE(3), \quad (2.2)$$

where $\mathbf{E}_j$ is the rigid body frame and $\mathbf{E}_i$ the reference frame. Define the following *elementary* rotation matrices:

$$R_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_\phi & -s_\phi \\ 0 & s_\phi & c_\phi \end{bmatrix}, R_y(\theta) = \begin{bmatrix} c_\theta & 0 & s_\theta \\ 0 & 1 & 0 \\ -s_\theta & 0 & c_\theta \end{bmatrix}, R_z(\psi) = \begin{bmatrix} c_\psi & -s_\psi & 0 \\ s_\psi & c_\psi & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

where s_* and c_* represent the sine and cosine functions of the subscript. These matrices represent rotations around the canonical axes of any orthogonal reference frame.

Remark 1. *Recall that any rotation matrix can be parametrized by some Euler angle convention [19], as for example the Roll, Pitch and Yaw (RPY) angles. Furthermore, since $SO(3)$ is diffeomorphic to the set of unit quaternions $\mathbb{H}^*$ [38], any rotation matrix can also be parametrized by two elements of $\mathbb{H}^*$, as explained in Appendix A.*

One way to represent a rotation matrix is using the Roll (ϕ), Pitch (θ) and Yaw

(ψ) (RPY) angles convention, which is a *minimal* representation for orientation:

$$\begin{aligned} R_{RPY}(\phi,\theta,\psi) &= R_z(\psi)\, R_y(\theta)\, R_x(\phi) \\ &= \begin{bmatrix} c_\psi\, c_\theta & c_\psi\, s_\phi\, s_\theta - c_\phi\, s_\psi & s_\phi\, s_\psi + c_\phi\, c_\psi\, s_\theta \\ c_\theta\, s_\psi & c_\phi\, c_\psi + s_\phi\, s_\psi\, s_\theta & c_\phi\, s_\psi\, s_\theta - c_\psi\, s_\phi \\ -s_\theta & c_\theta\, s_\phi & c_\phi\, c_\theta \end{bmatrix}. \end{aligned} \tag{2.3}$$

Note that we can also obtain the inverse mapping, that is, find the Euler angles $\{\phi,\theta,\psi\}$ given a rotation matrix R_{rpy} through the following relations:

$$R_{RPY} = \begin{bmatrix} c_\psi\, c_\theta & * & * \\ c_\theta\, s_\psi & * & * \\ -s_\theta & c_\theta\, s_\phi & c_\phi\, c_\theta \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix},$$

where

$$\phi = tg^{-1}\left(\frac{r_{32}}{r_{33}}\right), \qquad \theta = tg^{-1}\left(\frac{-r_{31}}{\sqrt{r_{32}^2 + r_{33}^2}}\right), \qquad \psi = tg^{-1}\left(\frac{r_{21}}{r_{11}}\right). \tag{2.4}$$

Remark 2. *Note that the positive sign on the square root in the pitch equation gives a pitch angle $\theta \in [-\pi/2, \pi/2)$, and not the complete interval $\theta \in [-\pi, \pi)$. Also, when $\theta = \pm\pi/2$ we have the singularity of the RPY representation, also known as gimbal lock or nadir/zenith configuration in the inertial stabilization literature. While this singularity can cause computational issues, it can be neglected depending on the application. For example, in applications of of low motion amplitude, where neither the vehicle nor the camera will achieve $\pm 90^\circ$ or values close to this (camera and/or vehicle pointing up or down).*

Finally, it is also useful to have the time derivative of the rotation matrix (2.3), which is given by:

$$\dot{R}_{RPY} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}, \tag{2.5}$$

where:

- $a_{11} = -s_\theta c_\psi \dot{\theta} - c_\theta s_\psi \dot{\psi}$,
- $a_{12} = (s_\phi s_\psi + c_\phi s_\theta c_\psi)\, \dot{\phi} + s_\phi c_\theta c_\psi \dot{\theta} - (c_\phi c_\psi + s_\phi s_\theta s_\psi)\, \dot{\psi}$,
- $a_{13} = (c_\phi s_\psi - s_\phi s_\theta c_\psi)\, \dot{\phi} + c_\phi c_\theta c_\psi \dot{\theta} + (s_\phi c_\psi - c_\phi s_\theta s_\psi)\, \dot{\psi}$,
- $a_{21} = -s_\theta s_\psi \dot{\theta} + c_\theta c_\psi \dot{\psi}$,

- $a_{22} = (c_\phi s_\theta s_\psi - s_\phi c_\psi)\,\dot{\phi} + s_\phi c_\theta s_\psi \dot{\theta} + (s_\phi s_\theta c_\psi - c_\phi s_\psi)\,\dot{\psi}$,
- $a_{23} = -(c_\phi c_\psi + s_\phi s_\theta s_\psi)\,\dot{\phi} + c_\phi c_\theta s_\psi \dot{\theta} + (s_\phi s_\psi + c_\phi s_\theta c_\psi)\,\dot{\psi}$,
- $a_{31} = -c_\theta \dot{\theta}$,
- $a_{32} = c_\phi c_\theta \dot{\phi} - s_\phi s_\theta \dot{\theta}$,
- $a_{33} = -s_\phi c_\theta \dot{\phi} - c_\phi s_\theta \dot{\theta}$.

Let $R \in GL(n)$. An interesting relation that shall be useful in the following formulations and holds for all invertible matrices is

$$\dot{R}^{-1} = -R^{-1}\,\dot{R}\,R^{-1}\,, \quad \forall R \in GL(n). \tag{2.6}$$

In particular, if $R \in SO(3)$, then $R^{-1} = R^{\mathsf{T}}$ and therefore

$$\dot{R}^{\mathsf{T}} = -R^{\mathsf{T}}\,\dot{R}\,R^{\mathsf{T}}\,, \quad \forall R \in SO(3). \tag{2.7}$$

The homogeneous transformation between frame $\mathbf{E}_i$ and the inertial frame, which gives the frame position and orientation in inertial space, is given by

$$g_i(\eta_i) = \begin{bmatrix} R_i & p_i \\ 0 & 1 \end{bmatrix} \in SE(3)\,, \tag{2.8}$$

where $\eta_i \in \mathbb{R}^6$ is a vector representing the local frame configuration (or pose) in $SE(3)$. Suppose that some local parametrization for $SO(3)$ used to represent the frame orientation, such as the RPY Euler angles. In this case, η_0 can be defined as

$$\eta_i = \begin{bmatrix} \eta_{i_1} \\ \eta_{i_2} \end{bmatrix} = \begin{bmatrix} x_i & y_i & z_i & \phi_i & \theta_i & \psi_i \end{bmatrix}^{\mathsf{T}} \in \mathbb{R}^6\,, \tag{2.9}$$

where $\eta_{i_1} = p_i \in \mathbb{R}^3$ is the vehicle inertial position and $\eta_{i_2} = \begin{bmatrix} \phi_i & \theta_i & \psi_i \end{bmatrix}^{\mathsf{T}}$ is the vector of Euler angles associated to the rotation matrix $R_i \in SO(3)$.

2.2.2 Velocity Twists and the Adjoint Map

A very useful concept when deriving kinematics and dynamics of single or multibody systems is the concept of velocity *twists*. Physically, twists represent the velocities of a rigid body and can be written as an $\mathbb{R}^6$ vector with the linear and angular velocities of a rigid body. However, we need to specify the frame in which the velocities are

represented. In robotics, there are three main ways to represent twists: in body, spatial or inertial coordinates.

Let $\vec{v}_{ij}$ and $\vec{\omega}_{ij}$ be the physical linear and angular velocities of an arbitrary moving frame $\mathbf{E}_j$ with respect to $\mathbf{E}_i$. They are represented by $v_{ij} \in \mathbb{R}^3$, $\omega_{ij} \in \mathbb{R}^3$ when written in the world frame $\mathbf{E}_w$ and by $v_{ij}^i \in \mathbb{R}^3$ ($v_{ij}^j \in \mathbb{R}^3$), $\omega_{ij}^i \in \mathbb{R}^3$ ($\omega_{ij}^j \in \mathbb{R}^3$) when written in the origin frame $\mathbf{E}_i$ (in its own body frame $\mathbf{E}_j$). Obviously, $v_{ij}^i = R_{ij}\, v_{ij}^j$ and $\omega_{ij}^i = R_{ij}\, \omega_{ij}^j$.

The *body velocity twist* $V_{ij}^j = \begin{bmatrix} (v_{ij}^j)^\mathsf{T} & (\omega_{ij}^j)^\mathsf{T} \end{bmatrix}^\mathsf{T}$ represents the linear and angular velocity of a rigid body with a frame $\mathbf{E}_j$ relative to another frame $\mathbf{E}_i$ as seen for an observer on the body frame $\mathbf{E}_j$. The matrix representation $\mathsf{se}(V_{ij}^j) \in se(3)$ of the body twist V_{ij}^j is:

$$\mathsf{se}(V_{ij}^j) = g_{ij}^{-1}\, \dot{g}_{ij} = \begin{bmatrix} \mathbf{S}(\omega_{ij}^j) & v_{ij}^j \\ 0 & 0 \end{bmatrix}, \qquad V_{ij}^j = \begin{bmatrix} R_{ij}^\mathsf{T}\, \dot{p}_{ij} \\ \mathbf{S}^{-1}(R_{ij}^\mathsf{T}\, \dot{R}_{ij}) \end{bmatrix}, \tag{2.10}$$

where $se(3)$ is the *Lie algebra* of $SE(3)$ [21] and $\mathbf{S}(\omega_{ij}^j) \in so(3)$ represents the $\mathbb{R}^3 \to \mathbb{R}^{3\times 3}$ skew-symmetric operator:

$$\omega_{ij}^j = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}, \qquad \mathbf{S}(\omega_{ij}^j) = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix}. \tag{2.11}$$

The *spatial velocity twist* $V_{ij}^S = \begin{bmatrix} (v_{ij}^S)^\mathsf{T} & (\omega_{ij}^S)^\mathsf{T} \end{bmatrix}^\mathsf{T}$ represents the linear and angular velocities of a local frame $\mathbf{E}_j$ attached to a rigid body with respect to frame $\mathbf{E}_i$ in spatial coordinates. The vector $\omega_{ij}^S = \omega_{ij}^i$ is the angular velocity of $\mathbf{E}_j$ relative to $\mathbf{E}_i$ observed from $\mathbf{E}_i$, and v_{ij}^S is the linear velocity of a point located at an extension of the rigid body when this point travels through the origin of $\mathbf{E}_i$. This non-intuitive interpretation for the linear part of the spatial twist was introduced in [21].

It can be shown that the matrix representation $\mathsf{se}(V_{ij}^S) \in se(3)$ of the spatial twist V_{ij}^S is given by:

$$\mathsf{se}(V_{ij}^S) = \dot{g}_{ij}\, g_{ij}^{-1} = \begin{bmatrix} \mathbf{S}(\omega_{ij}^S) & v_{ij}^S \\ 0 & 0 \end{bmatrix}, \qquad V_{ij}^S = \begin{bmatrix} -\dot{R}_{ij}\, R_{ij}^\mathsf{T}\, p_{ij} + \dot{p}_{ij} \\ \mathbf{S}^{-1}(\dot{R}_{ij}\, R_{ij}^\mathsf{T}) \end{bmatrix}. \tag{2.12}$$

The *inertial velocity twist* $V_{ij} = \begin{bmatrix} v_{ij}^\mathsf{T} & \omega_{ij}^\mathsf{T} \end{bmatrix}^\mathsf{T}$ is the velocity twist of the rigid body with a frame $\mathbf{E}_j$ relative to another frame $\mathbf{E}_i$, but represented in inertial coordinates

(seen from the world frame). It can be easily found by:

$$V_{ij} = \begin{bmatrix} R_{ij} & 0 \\ 0 & R_{ij} \end{bmatrix} V_{ij}^{j}\,, \qquad V_{ij} = \begin{bmatrix} \dot{p}_{ij} \\ R_{ij}\,\mathbf{S}^{-1}(R_{ij}^{\mathsf{T}}\,\dot{R}_{ij}) \end{bmatrix}. \tag{2.13}$$

Note that this is not the same as the spatial velocity twist, although it has a more intuitive interpretation than the spatial velocity twist.

With (2.10) and (2.12) we can establish an important kinematic relation between the velocities expressed in body and spatial coordinates. This relation is given by the **Adjoint map** Ad_{ij} as follows [21]:

$$V_{ij}^{S} = Ad_{g_{ij}} V_{ij}^{j}\,, \qquad Ad_{g_{ij}} = \begin{bmatrix} R_{ij} & \mathbf{S}(p_{ij}^{i})\,R_{ij} \\ 0 & R_{ij} \end{bmatrix}. \tag{2.14}$$

We can also get the inverse of the adjoint map:

$$V_{ij}^{j} = Ad_{g_{ij}}^{-1}\,V_{ij}^{S}\,, \qquad Ad_{g_{ij}}^{-1} = Ad_{g_{ij}^{-1}} = Ad_{g_{ji}} = \begin{bmatrix} R_{ij}^{\mathsf{T}} & -R_{ij}^{\mathsf{T}}\,\mathbf{S}(p_{ij}^{i}) \\ 0 & R_{ij}^{\mathsf{T}} \end{bmatrix}. \tag{2.15}$$

The two *absolute* body velocity twists (with respect to the world frame $\mathbf{E}_w$) associated to different frames $\mathbf{E}_i$, $\mathbf{E}_j$ located in the *same* rigid-body are also related through the constant adjoint map $Ad_{g_{ij}} \in \mathbb{R}^{6\times 6}$ by

$$V_{j}^{j} = Ad_{g_{ij}}^{-1} V_{i}^{i}\,. \tag{2.16}$$

On the other hand, if $\mathbf{E}_i$ and $\mathbf{E}_j$ are fixed to *different* rigid-bodies, their absolute body twists are related by the relative body twist velocity V_{ij}^{j} between the bodies, yielding

$$V_{j}^{j} = V_{ij}^{j} + Ad_{g_{ij}}^{-1}\,V_{i}^{i}\,, \tag{2.17}$$

and the inverse relation

$$V_{i}^{i} = Ad_{g_{ij}}(V_{ij}^{j} - V_{j}^{j})\,. \tag{2.18}$$

Another useful quantity is the *acceleration twist*, which here is defined simply as the time derivative of any velocity twist. Differentiating (2.17) with respect to time, we get:

$$\dot{V}_{j}^{j} = \dot{V}_{ij}^{j} + Ad_{g_{ij}}^{-1}\,\dot{V}_{i}^{i} + \dot{Ad}_{g_{ij}}^{-1}\,V_{i}^{i}\,, \tag{2.19}$$

where the time derivative of the adjoint matrix and its inverse are given by:

$$\dot{A}d_{g_{ij}} = \begin{bmatrix} \dot{R}_{ij} & \mathbf{S}(\dot{p}^i_{ij})\, R_{ij} + \mathbf{S}(p^i_{ij})\dot{R}_{ij} \\ 0 & \dot{R}_{ij} \end{bmatrix}, \tag{2.20}$$

$$\dot{A}d_{g_{ji}} = \begin{bmatrix} \dot{R}^{\mathsf{T}}_{ij} & -\dot{R}^{\mathsf{T}}_{ij}\, \mathbf{S}(p^i_{ij}) - R^{\mathsf{T}}_{ij}\, \mathbf{S}(\dot{p}^i_{ij}) \\ 0 & \dot{R}^{\mathsf{T}}_{ij} \end{bmatrix}. \tag{2.21}$$

Note that if $\mathbf{E}_i$ and $\mathbf{E}_j$ are located in the same rigid body, $V^j_{ij} = \dot{V}^j_{ij} = 0$ and $\dot{A}d_{g_{ji}} = \dot{A}d^{-1}_{g_{ij}}$ is a null matrix, resulting simply in the time derivative of (2.16) with a constant adjoint matrix:

$$\dot{V}^j_j = Ad^{-1}_{g_{ij}}\, \dot{V}^i_i\,. \tag{2.22}$$

It is useful to represent the frame velocities in inertial space. So, we have to derive a velocity mapping between the frame velocity twist $V^i_i \in \mathbb{R}^6$ with the time derivatives $\dot{\eta}_i \in \mathbb{R}^6$ of the frame position and orientation (represented by Euler angles) defined in (2.9). For linear velocities, the relation is trivial:

$$\dot{\eta}_{i_1} = R_i(\eta_{i_2}) v^i_i\,, \tag{2.23}$$

where $R_i(\eta_{i_2})$ is given by (2.3), in the case of RPY angles.

The relation between the body angular velocities ω^i_i and the time derivative $\dot{\eta}_{i_2}$ of the RPY Euler angles is given by the *representation Jacobian* $T_i(\eta_{i_2}) \in \mathbb{R}^{3\times 3}$. It is derived accounting that $\mathbf{S}(\omega^i_i) = R^{\mathsf{T}}_i\, \dot{R}_i$. For the *particular case* of the RPY representation, R_i and $\dot{R}_i$ are represented by (2.3) and (2.5), and thus

$$\omega^i_i = \mathbf{S}^{-1}(R^{\mathsf{T}}_i\, \dot{R}_i) = \begin{bmatrix} \dot{\phi}_i - s_{\theta_i}\, \dot{\psi}_i \\ c_{\theta_i}\, s_{\phi_i}\, \dot{\psi}_i + c_{\phi_i}\, \dot{\theta}_i \\ c_{\theta_i}\, c_{\phi_i}\, \dot{\psi}_i - s_{\phi_i}\, \dot{\theta}_i \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & -s_{\theta_i} \\ 0 & c_{\phi_i} & c_{\theta_i}\, s_{\phi_i} \\ 0 & -s_{\phi_i} & c_{\theta_i}\, c_{\phi_i} \end{bmatrix}}_{T_i(\eta_{i_2})} \underbrace{\begin{bmatrix} \dot{\phi}_i \\ \dot{\theta}_i \\ \dot{\psi}_i \end{bmatrix}}_{\dot{\eta}_{0_2}}$$

$$\omega^i_i = T_i(\eta_{i_2})\, \dot{\eta}_{i_2}\,. \tag{2.24}$$

The inverse of this mapping is given by:

$$\begin{bmatrix} \dot{\phi}_i \\ \dot{\theta}_i \\ \dot{\psi}_i \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & \frac{s_{\phi_i}\, s_{\theta_i}}{c_{\theta_i}} & \frac{c_{\phi_i}\, s_{\theta_i}}{c_{\theta_i}} \\ 0 & c_{\phi_i} & -s_{\phi_i} \\ 0 & \frac{s_{\phi_i}}{c_{\theta_i}} & \frac{c_{\phi_i}}{c_{\theta_i}} \end{bmatrix}}_{T^{-1}_i(\eta_{i_2})} \omega^i_i\,,$$

$$\dot{\eta}_{i_2} = T_i^{-1}(\eta_{i_2})\,\omega_i^i\,. \tag{2.25}$$

Remark 3. *Equation* (2.24) *is valid for any local representation of* $SO(3)$*, not only the RPY angles. However, the particular form of the Representation Jacobian matrix is dependent on the choice of Euler angles. Besides,* (2.25) *may not be valid in certain configurations, due to the presence of singularities in* $T_i(\eta_{i_2})$*. In the particular case of the RPY angles, these singularities occur at* $\theta_i = \pm\frac{\pi}{2}$*.*

Grouping (2.23) and (2.25) into a single relation gives:

$$\begin{bmatrix} \dot{\eta}_{i_1} \\ \dot{\eta}_{i_2} \end{bmatrix} = \underbrace{\begin{bmatrix} R_i(\eta_{i_2}) & 0 \\ 0 & T_i^{-1}(\eta_{i_2}) \end{bmatrix}}_{J_i^{-1}(\eta_{i_2})} \begin{bmatrix} v_i^i \\ \omega_i^i \end{bmatrix},$$

$$\dot{\eta}_i = J_i^{-1}(\eta_{i_2})\,V_i^i\,, \tag{2.26}$$

where $J_i^{-1}(\eta_{i_2}) \in \mathbb{R}^{6\times 6}$ is the Jacobian matrix that maps the body velocity twist to the time derivative of the pose. It is also possible to write the inverse mapping as

$$\begin{bmatrix} v_i^i \\ \omega_i^i \end{bmatrix} = \underbrace{\begin{bmatrix} R_i^{\mathsf{T}}(\eta_{i_2}) & 0 \\ 0 & T_i(\eta_{i_2}) \end{bmatrix}}_{J_i(\eta_{i_2})} \begin{bmatrix} \dot{\eta}_{i_1} \\ \dot{\eta}_{i_2} \end{bmatrix},$$

$$V_i^i = J_i(\eta_{i_2})\,\dot{\eta}_i\,. \tag{2.27}$$

Moreover, taking the time derivatives of $T_i^{-1}(\eta_{i_2})$ and $T_i(\eta_{i_2})$ given in (2.24) and (2.25):

$$\dot{T}_i^{-1}(\eta_{i_2},\dot{\eta}_{i_2}) = \begin{bmatrix} 0 & c_\phi t_\theta\dot{\phi} + \frac{s_\phi}{c_\theta^2}\dot{\theta} & -s_\phi t_\theta\dot{\phi} + \frac{c_\phi}{c_\theta^2}\dot{\theta} \\ 0 & -s_\phi\dot{\phi} & -c_\phi\dot{\phi} \\ 0 & \frac{c_\phi}{c_\theta}\dot{\phi} + \frac{s_\phi s_\theta}{c_\theta^2}\dot{\theta} & -\frac{s_\phi}{c_\theta}\dot{\phi} + \frac{c_\phi s_\theta}{c_\theta^2}\dot{\theta} \end{bmatrix}, \tag{2.28}$$

$$\dot{T}_i(\eta_{i_2},\dot{\eta}_{i_2}) = \begin{bmatrix} 0 & 0 & -c_\theta\dot{\theta} \\ 0 & -s_\phi\dot{\phi} & c_\phi c_\theta\dot{\phi} - s_\phi c_\theta\dot{\theta} \\ 0 & -c_\phi\dot{\phi} & -s_\phi c_\theta\dot{\phi} + c_\phi s_\theta\dot{\theta} \end{bmatrix}. \tag{2.29}$$

Note that, in (2.28), we take the inverse of T_i first and then derive it. A useful relation to be applied for (2.28) is, for any invertible matrix T:

$$\dot{T}^{-1} = -T^{-1}\,\dot{T}\,T^{-1}\,. \tag{2.30}$$

Now, it is easy to also find the time derivative of $J_i^{-1}(\eta_{i_2})$ grouping (2.5) with (2.28), and the time derivative of and $J_i(\eta_{i_2})$ grouping the transpose of (2.5) with

(2.29):

$$\dot{J}_i^{-1}(\eta_{i_2}, \dot{\eta}_{i_2}) = \begin{bmatrix} \dot{R}_i & 0 \\ 0 & \dot{T}_i^{-1} \end{bmatrix}, \tag{2.31}$$

$$\dot{J}_i(\eta_{i_2}, \dot{\eta}_{i_2}) = \begin{bmatrix} \dot{R}_i^{\mathsf{T}} & 0 \\ 0 & \dot{T}_0 \end{bmatrix} \tag{2.32}$$

Additionally, we need the relations between the vehicle body accelerations $\dot{V}_i^i$ and the second time derivative of the vehicle pose $\ddot{\eta}_i$. It is straightforward to derive this relation through the derivation in time of (2.26) and also get the inverse mappings, as follows:

$$\dot{V}_i^i = J_i(\eta_{i_2})\ddot{\eta}_i + \dot{J}_i(\eta_{i_2}, \dot{\eta}_{i_2})\dot{\eta}_i\,, \tag{2.33}$$

$$\ddot{\eta}_i = \dot{J}_i^{-1}(\eta_{i_2}, \dot{\eta}_{i_2})V_i^i + J_i^{-1}(\eta_{i_2})\dot{V}_i^i\,. \tag{2.34}$$

2.2.3 Platform Kinematics

The proposed inertial stabilization platform (a 3DOF gimbaled structure) mounted on a mobile base, such as a vehicle, can be characterized as a vehicle-manipulator system. A robotic manipulator is composed of a collection of rigid bodies, or links, whose relative motion is constrained by the admissible velocities of the joint connecting two consecutive rigid bodies.

Remark 4. *The links of the 3-DOF platform are rigid, each joint only allow 1DOF (Euclidean joints) to the link that follows it, and the manipulator structure is of open-chain (serial) type.*

The ISP can be seen as a manipulator with 3 links and 3 joints, which provide relative motion between adjacent links. The last link has the pointing sensor (camera) attached to it. The i^{th} link is coupled to the $(i+1)^{th}$ link by the 1DOF revolution joint $i+1$ coupled to a motor with its stator fixed on link i and rotor on link $i+1$, $i = 0, 1, 2$. Using this notation, link 0 actually represents the vehicle where the manipulator is mounted, which is a free rigid body with 6DOF, 3DOF for translation and 3DOF for rotation.

2.2.3.1 Forward Kinematics

In robotic manipulator systems, it is important to know the relation between the joint position configuration and the end-effector pose, which is given by *forward*

kinematics map. The forward kinematics on the i-th link frame $\mathbf{E}_i$ with respect to the vehicle frame $\mathbf{E}_0$ consists on the homogeneous transformation

$$g_{0i}(q, \Pi_g) = g_{01}\, g_{12} \ldots g_{i-1,i}\,. \tag{2.35}$$

One way to derive an explicit expression for the forward kinematics map is using the concept of **joint twists**. The *body joint twist* describes the allowed joint motion as seen from the frame attached to it.

For a prismatic joint, the body joint twist describes the admissible linear velocities, i.e., the direction for which motion is allowed. Similarly, for a revolution joint, the body joint twist describes the admissible direction of angular motion p_i^i, represented in the body frame $\mathbf{E}_i$.

Remark 5. *As the stabilization platform does not have prismatic joints, we will be addressing only the case of revolution joints in the following. However, it is very straightforward to derive the same equations for the prismatic joint case and these derivations can be found in [19].*

The *body joint twist* of the $(i+1)$-th *revolution* joint $(i = 0, 1, 2)$ is given by $(X_i^i)^\mathsf{T} = \begin{bmatrix}\mathbf{0}^\mathsf{T} & (h_{i+1}^i)^\mathsf{T}\end{bmatrix}$. The corresponding *spatial joint twists* are given by $X_i^S = Ad_{g_{0i}}\, X_i^i$:

$$X_i^S = \begin{bmatrix} v_i^S \\ \omega_i^S \end{bmatrix} = \begin{bmatrix} R_{0i} & \mathbf{S}(p_{0i})\, R_{0i} \\ 0 & R_{0i} \end{bmatrix} \begin{bmatrix} 0 \\ h_{i+1}^i \end{bmatrix} = \begin{bmatrix} -\mathbf{S}(h_{i+1})\, l_i \\ h_{i+1} \end{bmatrix}, \tag{2.36}$$

where $h_{i+1} = R_{0i}\, h_{i+1}^i$ and $l_i \in \mathbb{R}^3$ is any point on this axis represented in frame $\mathbf{E}_0$. This point is generally chosen to be the origin of $\mathbf{E}_i$.

Having defined the ISP joint twists, we can obtain the forward kinematics through the product of exponentials formula [21]:

$$g_{0i}(q) = \mathbf{E}_1\, \mathbf{E}_2 \ldots \mathbf{E}_i\, g_{0i}(0)\,, \tag{2.37}$$

where $\mathbf{E}_i = \mathbf{e}^{\mathsf{se}(X_i^S)\, q_i} \in SE(3)$ is the exponential map of joint i, given by [2]

$$\mathbf{E}_i = \begin{bmatrix} \mathbf{e}_i & (\mathbf{I}_3 - \mathbf{e}_i)\, \mathbf{S}(\omega_i^S)\, v_i^S + \omega_i^S\, (\omega_i^S)^\mathsf{T}\, v_i^S\, q_i \\ 0 & 1 \end{bmatrix}, \tag{2.38}$$

where $\mathbf{e}_i = \mathbf{e}^{\mathbf{S}(\omega_i^S)\, q_i} \in SO(3)$ is simply the exponential representation of the rotation matrix of joint i by q_i radians on axis $\omega_i^S \in \mathbb{R}^3$. This exponential can be computed

[2] For revolution joints only. For prismatic joints, the rotation matrix is identity and the translation vector is given by $v_i q_i$.

using the *Rodrigues' formula*:

$$\mathbf{e}_i = \mathbf{I}_3 + s_{q_i}\,\mathbf{S}(\omega_i^S) + (1 - c_{q_i})\,\mathsf{so}^2(\omega_i^S)\,. \tag{2.39}$$

Using (2.36) and (2.39) in (2.38), we simplify the exponential map expression to:

$$\mathbf{E}_i = \begin{bmatrix} \mathbf{I}_3 + s_{q_i}\,\mathbf{S}(\omega_i^S) + (1 - c_{q_i})\,\mathsf{so}^2(\omega_i^S) & \left(-s_{q_i}\,\mathbf{I}_3 + (c_{q_i} - 1)\,\mathbf{S}(\omega_i^S)\right)\,\mathbf{S}(\omega_i^S)\,l_i \\ \mathbf{0}^\mathsf{T} & 1 \end{bmatrix}, \tag{2.40}$$

considering that each joint i if of revolution type.

Using (2.35), the *forward kinematics* of the camera with respect to the vehicle can be expressed by

$$g_{0c}(q, \Pi_g) = g_{01}(q_1, h_1^0, p_{01}^0)\, g_{12}(q_2, h_2^1, p_{12}^1)\, g_{23}(q_3, h_3^2, p_{23}^2)\, g_{3c}\,, \tag{2.41}$$

where $g_{01}, g_{12}, g_{23} \in SE(3)$ are functions of the three joint angles $q_1, q_2, q_3 \in [-\pi, \pi]$ and of the joint axes h_{i+1}^i and joint displacements $p_{i,i+1}^i$ from $g_{i,i+1}$ $(i = 0, 1, 2)$. The components of these parameters forms $\Pi_g \in \mathbb{R}^{N_g}$, the vector of *geometric* parameters of the ISP. Lastly, $g_{3c} \in SE(3)$ is a constant transformation from the third frame $\mathbf{E}_3$ to the camera frame $\mathbf{E}_c$.

The *absolute* pose of the camera (with respect to the world frame) can then be expressed by

$$g_c(p_0, R_0, q) = g_0(R_0, p_0)\, g_{0c}(q, \Pi_g)\,, \tag{2.42}$$

where $p_0 \in \mathbb{R}^3$, $R_0 \in SO(3)$ are the position and rotation of the vehicle, respectively.

2.2.3.2 Differential Kinematics

The other main important problem in the kinematics of robotic manipulators is to find the mapping between the end-effector velocity with the joint velocities. This is the *differential kinematics* problem and the relation between these velocities is given by the **Geometric Jacobian** matrix, which is derived in this section using the concepts of joint twists.

In this case, the *body geometric Jacobian* $J_{0i}^i \in \mathbb{R}^{6\times n}$ associated to link i maps the ISP joint velocities $\dot{q}$ to the body twist coordinates V_{0i}^i by a linear relation

$$V_{0i}^i = J_{0i}^i(q, \Pi_g)\,\dot{q}\,. \tag{2.43}$$

The body geometric Jacobian expression can be derived by observing that:

$$\mathsf{se}(V_{0i}^i) = g_{0i}^{-1}(q,\Pi_g)\,\dot{g}_{0i}(q,\Pi_g)\,, \qquad \dot{g}_{0i}(q,\Pi_g) = \sum_{j=1}^{n}\left(\frac{\partial g_{0i}}{\partial q_j}\,\dot{q}_j\right)\,.$$

Then, the body geometric Jacobian is found by collecting in its columns the joint twists $X_i^\dagger(q)$ observed from the local frame $\mathbf{E}_i$. Each column maps the contribution of each joint velocity to the velocities of $\mathbf{E}_i$, in body coordinates:

$$J_{0i}^i(q,\Pi_g) = \begin{bmatrix} X_1^\dagger & X_2^\dagger & \cdots & X_i^\dagger & 0_{6\times(3-i)} \end{bmatrix} \tag{2.44}$$

$$= \begin{bmatrix} Ad_{g_{1i}}^{-1}\,X_1^1 & Ad_{g_{2i}}^{-1}\,X_2^2 & \cdots & Ad_{g_{ii}}^{-1}\,X_i^i & 0_{6\times(3-i)} \end{bmatrix}\,. \tag{2.45}$$

Similarly, we can find the *spatial geometric Jacobian*, which maps joint velocities to the spatial link velocity twist. The Jacobian columns correspond to the joint twists written in spatial coordinates with respect to the base frame $\mathbf{E}_0$.

$$V_{0i}^S = J_{0i}^S(q,\Pi_g)\,\dot{q}\,, \tag{2.46}$$

and the spatial link Jacobian can be computed by

$$J_{0i}^S(q,\Pi_g) = \begin{bmatrix} X_1^S & X_2^S & \cdots & X_i^S \end{bmatrix} \tag{2.47}$$

$$= \begin{bmatrix} Ad_{g_{01}}\,X_1^1 & Ad_{g_{02}}\,X_2^2 & \cdots & Ad_{g_{0i}}\,X_i^i & 0_{6\times(3-i)} \end{bmatrix}\,. \tag{2.48}$$

Note that X_i' is the twist of joint i seen from $\mathbf{E}_0$ and depends only on the position of the previous joints $(q_1,\ldots,q_{i-1})$. So, the the adjoint map $Ad_{g_{0i}}$ depends only on $(q_1,\ldots,q_{i-1})$ in this case

$$X_i' = Ad_{g_{0i}}(q_1,\ldots,q_{i-1},\Pi_g)\,X_i^i\,.$$

Due to (2.43), (2.46) and (2.14), the relation between the body and spatial link Jacobians is given by

$$J_{0i}^S = Ad_{g_{0i}}\,J_{0i}^c\,. \tag{2.49}$$

Sometimes, only the linear or angular parts of the geometric Jacobians are needed. Therefore, they can be partitioned as

$$J_{0i}^i(q) = \begin{bmatrix} J_{0i_1}^i \\ J_{0i_2}^i \end{bmatrix}\,, \quad J_{0i}^S(q) = \begin{bmatrix} J_{0i_1}^S \\ J_{0i_2}^S \end{bmatrix}\,. \tag{2.50}$$

where $J_{0i_1}^i, J_{0i_2}^i, J_{0i_1}^S, J_{0i_2}^S \in \mathbb{R}^{3\times n}$.

Using (2.17) and (2.43), the absolute body velocity twist $V_i^i \in \mathbb{R}^6$ associated to the i-th link of the ISP can be expressed by

$$V_i^i = J_{0i}^i(q, \Pi_g)\, \dot{q} + Ad_{g_{0i}}^{-1}(q, \Pi_g)\, V_0^0\,, \tag{2.51}$$

where $V_0^0 \in \mathbb{R}^6$ is the base body velocity twist. The *acceleration* twists can also be computed by means of

$$\dot{V}_i^i = J_{0i}^i(q, \Pi_g)\, \ddot{q} + \dot{J}_{0i}^i(q, \dot{q}, \Pi_g)\, \dot{q} + Ad_{g_{0i}}^{-1}(q, \Pi_g)\, \dot{V}_0^0 + \dot{Ad}_{g_{0i}}^{-1}(q, \dot{q}, \Pi_g)\, V_0^0\,, \tag{2.52}$$

where $\dot{V}_0^0 \in \mathbb{R}^6$ is the base body acceleration twist.

Note that, when applied to the *camera* frame $\mathbf{E}_c$ on the ISP last link, the *angular parts* of (2.51) and (2.52) are

$$\omega_c^c = J_{0c_2}^c(q, \Pi_g)\, \dot{q} + \omega_0^c\,, \tag{2.53}$$

$$\dot{\omega}_c^c = J_{0c_2}^c(q, \Pi_g)\, \ddot{q} + \dot{J}_{0c_2}^c(q, \dot{q}, \Pi_g)\, \dot{q} + \dot{\omega}_0^c\,, \tag{2.54}$$

Note how the vehicle angular velocity and acceleration $\omega_0^0, \dot{\omega}_0^0 \in \mathbb{R}^3$ expressed on $\mathbf{E}_0$ emerges from the last two terms of (2.51) and (2.52), respectively. The camera body geometric Jacobian is expressed by

$$J_{0c}^c(q, \Pi_g) = \begin{bmatrix} X_1^\dagger & X_2^\dagger & X_3^\dagger \end{bmatrix} \tag{2.55}$$

$$= \begin{bmatrix} Ad_{g_{1c}}^{-1} X_1^1 & Ad_{g_{2c}}^{-1} X_2^2 & Ad_{g_{3c}}^{-1} X_3^3 \end{bmatrix}. \tag{2.56}$$

Due to the ISP structure, $X_1^1 = \begin{bmatrix} 0^\mathsf{T} & z_0^\mathsf{T} \end{bmatrix}^\mathsf{T}$, $X_2^2 = \begin{bmatrix} 0^\mathsf{T} & y_0^\mathsf{T} \end{bmatrix}^\mathsf{T}$ and $X_3^3 = \begin{bmatrix} 0^\mathsf{T} & x_0^\mathsf{T} \end{bmatrix}^\mathsf{T}$, representing the twists associated to each joint of the ISP.

An important algebraic property is the *linearity* of (2.53) and (2.54) with respect to the *geometric* parameters of the ISP [20]:

$$\omega_c^c = W_\omega(q, \dot{q}, \omega_0^c)\, \Pi_g\,, \tag{2.57}$$

$$\dot{\omega}_c^c = \dot{W}_\omega(q, \dot{q}, \ddot{q}, \omega_0^c, \dot{\omega}_0^c)\, \Pi_g\,. \tag{2.58}$$

where $W_\omega \in \mathbb{R}^{3 \times N_g}$ is a *kinematic regressor*.

Next, applying (2.34) to $\mathbf{E}_c$ and taking the angular part yields

$$\ddot{\eta}_{c_2} = T_c^{-1}(\eta_{c_2})\, \dot{\omega}_c^c + \dot{T}_c^{-1}(\eta_{c_2}, \dot{\eta}_{c_2})\, \omega_c^c\,. \tag{2.59}$$

Substituting (2.53) and (2.54) in (2.59), it is possible to write it with respect to

the ISP variables and the ship motion as:

$$\ddot{\eta}_{c_2} = \underbrace{T_c^{-1}\, J^c_{0c_2}}_{J_q}\, \ddot{q} + \underbrace{\left(T_c^{-1}\, \dot{J}^c_{0c_2} + \dot{T}_c^{-1}\, J^c_{0c_2}\right)}_{L_q}\, \dot{q} + \underbrace{T_c^{-1}\, R^{\mathsf{T}}_{0c}}_{J_\omega}\, \dot{\omega}^0_0 + \underbrace{\dot{T}_c^{-1}\, R^{\mathsf{T}}_{0c}}_{L_\omega}\, \omega^0_0\,, \qquad (2.60)$$

where matrices J_q and J_ω are dependent on q, η_{c_2} and Π_g, while L_q, L_ω are also dependent on $\dot{q}$ and $\dot{\eta}_{c_2}$.

In the same way than in (2.58), (2.60) is also linear with respect to the *geometric* parameters:

$$\ddot{\eta}_{c_2} = W_\eta(q, \dot{q}, \ddot{q}, \eta_{c_2}, \dot{\eta}_{c_2}, \omega^0_0, \dot{\omega}^0_0)\, \Pi_g\,, \qquad (2.61)$$

where $W_\eta \in \mathbb{R}^{3\times N_g}$ is a *kinematic regressor*, and N_g is the number of geometric parameters.

2.2.3.3 Kinematic Algorithms

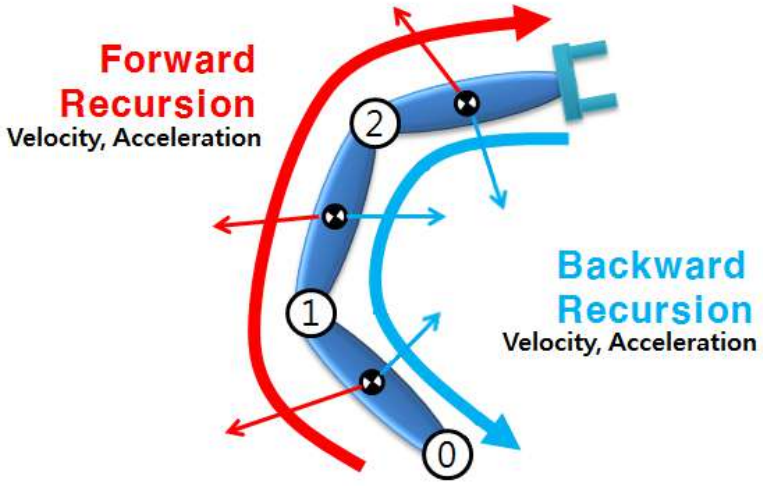


Figure 2.2: Forward and backward recursion for computing pose, velocities and accelerations of a serial robot manipulator. Edited from http://www.matlabinuse.com/Mastering_MATLAB/11349.

Next, given the pose of frame $\mathbf{E}_i$ $g_i \in SE(3)$ and the frame velocity and acceleration twists V^i_i, $\dot{V}^i_i$, it is possible to compute all other poses g_k, velocities V^k_k and accelerations $\dot{V}^k_k$ of to each other frame $\mathbf{E}_k$ $(k \neq i)$ by means of an iterative algorithm. It consists in propagating the poses and/or body velocity/acceleration twists of each link frame $\mathbf{E}_i$ through the system, obtaining all g_k, V^k_k, $\dot{V}^k_k$ $(k = 1, 2, 3)$ associated to all rigid bodies of the ISP.

Figure 2.2 illustrates Algorithm 1 and Algorithm 2 for a robot manipulator, which in this case, is the ISP itself.

Algorithm 1 (Pose Propagation)**.** *The algorithm is initialized with the configuration of the i-th frame* $\mathbf{E}_i$*,* $p_i \in \mathbb{R}^3$ *and* $r_i \in \mathbb{H}^*$*. The propagation algorithm is carried out in two steps:*

1. ***Upward propagation****: starting from index* $k = i$ *till* $k = n$*, the configuration of each frame can be computed by*

$$p_{k+1} = p_k + R_k \, p^k_{k,k+1} \,,$$
$$r_{k+1} = r_k \circ r^k_{k+1} \,.$$

2. ***Backward propagation****: starting from index* $k = i$ *till* $k = 0$*, the configuration of each frame can be computed by*

$$p_{k-1} = p_k - R_{k-1} \, p^{k-1}_{k-1,k} \,,$$
$$r_{k-1} = r_k \circ r^k_{k-1} \,.$$

Here, R_k *is computed from* r_k *in* (A.12) *(*$k = 1, 2, \ldots n$*). The camera pose is*

$$p_c = p_n + R_n \, p^n_{nc} \,,$$
$$r_c = r_n \circ r^n_c \,.$$

If the inertial pose of any frame $\mathbf{E}_0$ rigidly attached to the vehicle and the joint angles $q \in \mathbb{R}^n$ are fully known, Algorithm 1 can be used to compute all homogeneous transformations $g_i \in SE(3)$ for each coordinate system $\mathbf{E}_i$ $(i = 1, 2, 3)$ associated to the ISP links and for the ISP camera frame $\mathbf{E}_c$.

The next algorithm is able to compute all velocity and acceleration twists associated to each one of the ISP coordinate systems, given that the velocity and acceleration twists of any frame $\mathbf{E}_i$ and the joint angle, velocities and accelerations $q, \dot{q}, \ddot{q} \in \mathbb{R}^n$ are known.

Algorithm 2 (Propagation of Velocities and Accelerations)**.** *The algorithm is initialized with the motion variables of the i-th link frame* $\mathbf{E}_i$*. Given that the mechanism contains only 1DOF rotational joints, the algorithm can be described in two steps:*

1. ***Upward propagation****: starting from index* $k = i$ *til* $k = n - 1$*, the velocities and accelerations of the links can be computed by*

$$V^{k+1}_{k+1} = \Omega^{\mathsf{T}}_{k,k+1} \left(\Phi_{k+1,k} \, V^k_k + H_{k+1} \, \dot{q}_{k+1} \right) ,$$
$$\dot{V}^{k+1}_{k+1} = \Omega^{\mathsf{T}}_{k,k+1} \left(\Phi_{k+1,k} \, \dot{V}^k_k + H_{k+1} \, \ddot{q}_{k+1} + A_{k+1} \, \dot{q}_{k+1} \right) .$$

2. ***Backward propagation****: starting from index $k = i$ til $k = 0$, the velocities and accelerations of the links can be computed by*

$$V_{k-1}^{k-1} = \Phi_{k,k-1}^{-1} \left(\Omega_{k-1,k} \, V_k^k - H_k \, \dot{q}_k \right)$$
$$\dot{V}_{k-1}^{k-1} = \Phi_{k,k-1}^{-1} \left(\Omega_{k-1,k} \, \dot{V}_k^k - H_k \, \ddot{q}_k - A_k \, \dot{q}_k \right).$$

with matrices given by

$$\Phi_{k+1,k} = \begin{bmatrix} \mathsf{I}_3 & -\mathsf{so}(p_{k,k+1}^k) \\ 0 & \mathsf{I}_3 \end{bmatrix}, \quad \Phi_{k+1,k}^{-1} = \begin{bmatrix} \mathsf{I}_3 & \mathsf{so}(p_{k,k+1}^k) \\ 0 & \mathsf{I}_3 \end{bmatrix},$$
$$H_{k+1}^{\mathsf{T}} = \begin{bmatrix} 0^{\mathsf{T}} (h_{k+1}^k)^{\mathsf{T}} \end{bmatrix} \quad , \quad \Omega_{k,k+1} = \begin{bmatrix} R_{k,k+1} & 0 \\ 0 & R_{k,k+1} \end{bmatrix},$$
$$A_{k+1} = \begin{bmatrix} \mathsf{so}(v_k^k + \mathsf{so}(\omega_k^k) \, p_{k,k+1}^k) \, h_{k+1}^k \\ \mathsf{so}(\omega_k^k) \, h_{k+1}^k \end{bmatrix},$$

where $R_{k,k+1} \in SO(3)$ are computed from $r_{k+1}^k = \left\{ cos\left(\frac{1}{2} q_{k+1}\right), h_{k+1}^k \, sin\left(\frac{1}{2} q_{k+1}\right) \right\} \in \mathbb{H}^*$ using (A.12)*. The camera twists can be computed by*

$$V_c^c = Ad_{g_{cn}} \, V_n^n,$$
$$\dot{V}_c^c = Ad_{g_{cn}} \, \dot{V}_n^n,$$

where $g_{cn} \in SE(3)$ is a constant homogeneous transformation.

This algorithm can be used to compute not only every velocity and acceleration twist of the ISP, but also the geometric Jacobian matrices and some other useful terms.

Algorithm 3 (Computation of Jacobian Matrices)**.** *The j-th column of the body Jacobian matrix $J_{0i}^i(q)$ $(0 < j \leq i)$ is equal to V_j^j, the velocity twist obtained by executing Algorithm 2 starting from $\mathbf{E}_0$ with $V_0^0 = \dot{V}_0^0 = 0$, $\dot{q}_j = 1$ and all remaining joint velocities equal to zero. Therefore, $J_{0i}^i \in \mathbb{R}^{6 \times n}$ can be fully computed (column by column) by executing Algorithm 2 n times, where n is the number of ISP joints.*

Algorithm 4 (Computation of term $\dot{J}_{0i}^i(q, \dot{q}) \, \dot{q}$)**.** *The term $\dot{J}_{0i}^i(q, \dot{q}) \, \dot{q} \in \mathbb{R}^6$ is equal to $\dot{V}_i^i$, the acceleration twist obtained by executing Algorithm 2 starting from $\mathbf{E}_0$ with $V_0^0 = \dot{V}_0^0 = 0$, $\ddot{q} = 0$.*

Remark 6. *Note that the time derivative of the body link geometric Jacobian $\dot{J}_{0i}^i(q, \dot{q}) \in \mathbb{R}^{6 \times n}$ cannot be computed by the same method of Algorithm 3, because it is also dependent on $\dot{q} \in \mathbb{R}^n$.*

2.2.4 LOS Kinematics

In order to obtain the desired camera orientation, we must derive the relations between the target frame $\mathbf{E}_t$ motion and the camera frame $\mathbf{E}_c$ motion. These geometric relations are also known as the *LOS kinematics*. They are needed to compute the camera orientation, angular velocities and angular accelerations that guarantee that it remains always pointing in the direction of the target.

Suppose that the position vector of the target frame $\mathbf{E}_t$ and its first and second time derivatives are given by

$$p_t = \begin{bmatrix} x_t & y_t & z_t \end{bmatrix}^\mathsf{T}, \quad \dot{p}_t = \begin{bmatrix} \dot{x}_t & \dot{y}_t & \dot{z}_t \end{bmatrix}^\mathsf{T}, \quad \ddot{p}_t = \begin{bmatrix} \ddot{x}_t & \ddot{y}_t & \ddot{z}_t \end{bmatrix}^\mathsf{T}. \tag{2.62}$$

Moreover, define the inertial LOS errors for position, velocity and acceleration as

$$p_{ct} = p_t - p_c = \begin{bmatrix} x_{ct} & y_{ct} & z_{ct} \end{bmatrix}^\mathsf{T} = \begin{bmatrix} x_t - x_c \\ y_t - y_c \\ z_t - z_c \end{bmatrix}, \tag{2.63}$$

$$\dot{p}_{ct} = \dot{p}_t - \dot{p}_c = \begin{bmatrix} \dot{x}_{ct} & \dot{y}_{ct} & \dot{z}_{ct} \end{bmatrix}^\mathsf{T} = \begin{bmatrix} \dot{x}_t - \dot{x}_c \\ \dot{y}_t - \dot{y}_c \\ \dot{z}_t - \dot{z}_{0c} \end{bmatrix}, \tag{2.64}$$

$$\ddot{p}_{ct} = \ddot{p}_t - \ddot{p}_c = \begin{bmatrix} \ddot{x}_{ct} & \ddot{y}_{ct} & \ddot{z}_{ct} \end{bmatrix}^\mathsf{T} = \begin{bmatrix} \ddot{x}_t - \ddot{x}_c \\ \ddot{y}_t - \ddot{y}_c \\ \ddot{z}_t - \ddot{z}_c \end{bmatrix}. \tag{2.65}$$

Consider that the camera is correctly pointing into the target direction, and recall that the homogeneous transformation between the target frame $\mathbf{E}_t$ and the inertial frame is given by $g_t = g_c\, g_{ct} \in SE(3)$. Taking only the translation part:

$$p_t = p_c + R_{c_d}\, p_{ct}^c, \qquad R_{c_d} = \begin{bmatrix} x_{c_d} & y_{c_d} & z_{c_d} \end{bmatrix}, \tag{2.66}$$

where $R_{c_d} \in SO(3)$ is the *desired* camera rotation matrix. Given that the x-axis direction of the camera frame $\mathbf{E}_c$ is aligned with the camera optical axis, the relation that ensures the correct aimpoint is given by

$$p_{ct}^c = \|p_{ct}\|\; x_c^c, \qquad x_c^c = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^\mathsf{T}. \tag{2.67}$$

The interpretation is very straightforward: to ensure that the LOS coincides with the aimpoint, the translational LOS error ($p_{ct}^c \in \mathbb{R}^3$) must point in the same direction of the camera optical axis. Then, substituting (2.67) into (2.66), we get the *pointing*

condition

$$x_{c_d} = \frac{p_{ct}}{\|p_{ct}\|}\,. \tag{2.68}$$

Note that the *pointing condition* (2.68) only ensures the correct aimpoint, but it does not restrict the *camera tilt* with respect to the horizon. Although aligned to the LOS pointing vector, $\mathbf{E}_c$ still have an infinite possible number of configurations, since it can rotate freely around the camera axis.

The desired camera roll orientation or *tilt* is actually a project requirement. Specially in the case of imaging systems, it must be aligned to the horizon to guarantee a satisfactory performance. Therefore, the camera roll orientation and its time derivatives must be zero ($\phi_c = \dot{\phi}_c = \ddot{\phi}_c = 0$). This condition can be ensured by guaranteeing that the y-axis direction of $\mathbf{E}_c$ is perpendicular to both the pointing vector p_{ct} and to the vertical z-axis direction. From this, we conclude that

$$y_{c_d} = \frac{z_0 \times p_{ct}}{\|z_0 \times p_{ct}\|}\,. \tag{2.69}$$

Finally, the third and last column of R_{c_d} can be computed by

$$z_{c_d} = x_{c_d} \times y_{c_d}\,. \tag{2.70}$$

Then, the mapping from $R_{c_d} \in SO(3)$ to $r_{c_d} \in \mathbb{H}^*$ is relatively straightforward. For instance, in [20],

For mobile target tracking, the angular velocity and acceleration references for the camera are

$$\mathbf{S}(\omega^c_{c_d}) = R^{\mathsf{T}}_{c_d}\dot{R}_{c_d}\,, \tag{2.71}$$

$$\mathbf{S}(\dot{\omega}^c_{c_d}) = \dot{R}^{\mathsf{T}}_{c_d}\dot{R}_{c_d} + R^{\mathsf{T}}_{c_d}\ddot{R}_{c_d}\,, \tag{2.72}$$

where $\dot{R}_{c_d}$, $\ddot{R}_{c_d}$ can be computed from the time derivatives of x_{c_d}, y_{c_d}, and z_{c_d}, which are dependent on $\dot{p}_{ct}$ and $\ddot{p}_{ct}$. These derivatives can be computed from the formulas:

$$\frac{d}{dt}\left(\frac{x}{\|x\|}\right) = A(x)\,\dot{x}\,, \tag{2.73}$$

$$\frac{d^2}{dt^2}\left(\frac{x}{\|x\|}\right) = A(x)\,\ddot{x} + \dot{A}(x,\dot{x})\,\dot{x}\,, \tag{2.74}$$

$$A(x) = \frac{x^{\mathsf{T}} x \, \mathsf{I}_3 - x x^{\mathsf{T}}}{\|x\|^3} \,,$$
$$\dot{A}(x) = \frac{1}{\|x\|^3} \left(2\, x^{\mathsf{T}} \dot{x} \, \mathsf{I}_3 - \dot{x} x^{\mathsf{T}} - x \dot{x}^{\mathsf{T}} - \frac{3\, x^{\mathsf{T}} \dot{x}}{\|x\|} A(x) \right) ,$$

which are obviously not defined for $x = 0$. Using (2.73) and (2.74), the first and second time derivatives of the the first and second columns of R_{c_d} are given by

$$\dot{x}_{c_d} = \frac{d}{dt} \left(\frac{x}{\|x\|} \right) \Big|_{x = p_{ct}} \quad , \quad \ddot{x}_{c_d} = \frac{d^2}{dt^2} \left(\frac{x}{\|x\|} \right) \Big|_{x = p_{ct}} \quad ,$$
$$\dot{y}_{c_d} = \frac{d}{dt} \left(\frac{x}{\|x\|} \right) \Big|_{x = z_0 \times p_{ct}} , \quad \ddot{y}_{c_d} = \frac{d^2}{dt^2} \left(\frac{x}{\|x\|} \right) \Big|_{x = z_0 \times p_{ct}} .$$

Additionally, due to (2.70), the time derivatives of the the third column of R_{c_d} is

$$\dot{z}_{c_d} = \dot{x}_{c_d} \times y_{c_d} + x_{c_d} \times \dot{y}_{c_d} \,,$$
$$\ddot{z}_{c_d} = \ddot{x}_{c_d} \times y_{c_d} + 2\, \dot{x}_{c_d} \times \dot{y}_{c_d} + x_{c_d} \times \ddot{y}_{c_d} \,.$$

Remark 7. *In fact, due to* (2.73) *and* (2.71)*, the desired angular velocity* $\omega^c_{c_d} \in \mathbb{R}^3$ *is linear with respect to* $\dot{p}_{ct} \in \mathbb{R}^3$*:*

$$\omega^c_{c_d} = Z(p_{ct})\, \dot{p}_{ct} \,, \tag{2.75}$$

where $Z(p_{ct}) \in \mathbb{R}^{3\times 3}$ *is a regressor matrix. The desired angular acceleration* $\dot{\omega}^c_{c_d}$ *can be written as*

$$\dot{\omega}^c_{c_d} = Z(p_{ct})\, \ddot{p}_{ct} + \dot{Z}(p_{ct}, \dot{p}_{ct})\, \dot{p}_{ct} \,. \tag{2.76}$$

Next, suppose that the RPY angles are chosen as a minimum representation for orientation. Then, by (2.66), (2.67) and using (2.3) for $R_{c_d} = R_{RPY}(\phi_{c_d}, \theta_{c_d}, \psi_{c_d})$:

$$x_{ct} = \|p_{ct}\| \; cos(\psi_{c_d})\, cos(\theta_{c_d}) \tag{2.77}$$
$$y_{ct} = \|p_{ct}\| \; sin(\psi_{c_d})\, cos(\theta_{c_d}) \tag{2.78}$$
$$z_{ct} = - \|p_{ct}\| \; sin(\theta_{c_d}) \,. \tag{2.79}$$

We have already shown that the condition (2.69) for the camera roll results in $\phi_{c_d} = \dot{\phi}_{c_d} = \ddot{\phi}_{c_d} = 0$. The pitch and yaw references can be computed after manipulating

equations (2.77), (2.78), (2.79) and their time-derivatives:

$$\begin{cases} \theta_{c_d} &= tg^{-1}\left(\dfrac{-z_{ct}}{\sqrt{t_0}}\right), \\ \dot{\theta}_{c_d} &= \dfrac{z_{ct}\,t_2 - \dot{z}_{ct}\,t_0}{\sqrt{t_0}\,t_1}, \\ \ddot{\theta}_{c_d} &= \dfrac{(\dot{z}_{ct}\,t_0 - z_{ct}\,t_2)\,(3\,t_0\,t_2 + z_{ct}^2\,t_2 + 2\,t_0\,z_{ct}\,\dot{z}_{ct})}{t_0\,\sqrt{t_0}\,t_1^2} + \dots \\ & \dots + \dfrac{z_{ct}\,(x_{ct}\,\ddot{x}_{ct} + y_{ct}\,\ddot{y}_{ct} + \dot{x}_{ct}^2 + \dot{y}_{ct}^2) - \ddot{z}_{ct}\,t_0 - \dot{z}_{ct}\,t_2}{\sqrt{t_0}\,t_1}; \end{cases} \tag{2.80}$$

$$\begin{cases} \psi_{c_d} &= tg^{-1}\left(\dfrac{y_{ct}}{x_{ct}}\right), \\ \dot{\psi}_{c_d} &= \dfrac{t_3}{t_0}, \\ \ddot{\psi}_{c_d} &= \dfrac{(x_{ct}\,\ddot{y}_{ct} - y_{ct}\,\ddot{x}_{ct})\,t_0 - 2\,t_2\,t_3}{t_0^2}, \end{cases} \tag{2.81}$$

$$t_0 \coloneqq x_{ct}^2 + y_{ct}^2, \qquad t_2 \coloneqq x_{ct}\,\dot{x}_{ct} + y_{ct}\,\dot{y}_{ct},$$
$$t_1 \coloneqq x_{ct}^2 + y_{ct}^2 + z_{et}^2, \qquad t_3 \coloneqq x_{ct}\,\dot{y}_{ct} - y_{ct}\,\dot{x}_{ct}.$$

Notice that the configuration known in the literature as the *gimbal lock*, or the RPY representation *singularity* happens in (2.80) when $t_0 = 0$, which means that the pitch angle θ_{c_d} reached $\pm 90°$.

Lastly, consider the that the camera absolute orientation $R_{c_d} \in SO(3)$ is expressed in terms of *unit quaternions*. This non-minimal representation is advantageous, since it does not suffer from representation singularities as the RPY angles, for example. In this case, the corresponding quaternion reference $r_{c_d} \in \mathbb{H}^*$ can be directly computed using (A.14). If necessary, the first and second time derivatives of the quaternion reference r_{c_d} can be related to the body angular velocity and acceleration by the lower part of (A.18).

2.3 Dynamic Modeling of Inertial Stabilization Systems

In this section, we develop the dynamic model for the ISP mechanism mounted on a general mobile base using the vehicle-manipulator system (VMS) framework, developed in [19], [39] and [40]. The dynamic modeling takes into account the forces

and torques that act on its structure, such as driver motor torques, static/dynamic friction and the coupling forces among the links, and the effects of these efforts in the system motion.

The dynamic model is basically a differential equation that drives the temporal evolution of the ISP and vehicle motion. As a mechanical structure, it must follow the general laws of classical mechanics, that can be expressed by the *Newton-Euler*, *Lagrangian* or *Hamiltonian* framework. In this work, the Newton-Euler and Lagrangian formulation are covered.

The ISP structure can be modeled as a robot manipulator composed of rotational 1DOF joints and rigid links installed on a rigid, moving, non-inertial base (vehicle) that is free to move in $SE(3)$. Electrical motors are the main source of motion effort, providing the necessary joint torques for the ISP operation.

In this work, the electrical model associated to the motors is neglected. We focus on the mechanical modeling of the ISP only, due to the availability of high performance motors and motor drivers in the market. These drivers often allow the effects associated to the electrical model of the motors to be negligible, such as motor response time and magnetic cogging.

2.3.1 Lagrangian Dynamics of Vehicle-Manipulator Systems

The Euler-Lagrange equations of motion for a mechanical system with N generalized coordinates (expressed by $x \in \mathbb{R}^N$), total kinetic energy $\mathcal{K}(x, \dot{x})$ and potential energy $\mathcal{U}(x)$ are given in vector form by

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{x}}\right) - \frac{\partial \mathcal{L}}{\partial x} = \bar{\tau}\,, \tag{2.82}$$

where $\mathcal{L} = \mathcal{K} - \mathcal{U}$ is the system *Lagrangian* function and $\bar{\tau} \in \mathbb{R}^N$ is the vector of generalized forces collocated with $\dot{x} \in \mathbb{R}^N$.

In [40], a procedure to develop an expression for the equations of motion of a general multibody system with *quasi-velocities* is presented. The quasi-velocities are variables that uniquely describe the system velocity, but are not necessarily obtained by the time derivative of the position variables. Instead, they are expressed by the linear relation

$$v \coloneqq S(x)\,\dot{x}\,. \tag{2.83}$$

Remark 8. *Notice that, in general, the generalized forces collocated to the quasi-*

velocities $v \in \mathbb{R}^N$ *are going to be different than the* $\bar{\tau} \in \mathbb{R}^N$*, due to* (2.83)*.*

Then, we present the following procedure to find $\tau \in \mathbb{R}^N$, the generalized forces collocated to the *quasi-velocities* $v \in \mathbb{R}^N$, instead of $\dot{x} \in \mathbb{R}^N$. The infinitesimal work done in the system is

$$dW = (\bar{\tau}^{\mathsf{T}}\, \dot{x})\, dt = (\tau^{\mathsf{T}}\, v)\, dt\,, \tag{2.84}$$

Then, from (2.83) and (2.84), $\bar{\tau}$ and τ are related by

$$\bar{\tau} = S^{\mathsf{T}}(x)\tau\,. \tag{2.85}$$

If the quasi-velocity $v \in \mathbb{R}^N$ represents a *physical velocity*, then the kinetic energy of the system $\mathcal{K}(x, v)$ is given by a quadratic function

$$\mathcal{K}(x, v) = \frac{1}{2}\, v^{\mathsf{T}}\, M(x)\, v\,, \tag{2.86}$$

where $M(x) \in \mathbb{R}^{N\times N}$ is a symmetric, positive-definite inertia matrix, which is configuration-dependent. Then, the Lagrangian of a multibody system with quasi-velocities $v \in \mathbb{R}^N$ given by (2.83) be

$$\mathcal{L}(x, v) = \frac{1}{2}\, v^{\mathsf{T}}\, M(x)\, v - \mathcal{U}(x)\,. \tag{2.87}$$

Lemma 1. *The equations of motion of a multibody system with quasi-velocities* $v \in \mathbb{R}^N$ *defined in* (2.83)*, generalized forces* $\tau \in \mathbb{R}^N$ *and Lagrangian function* (2.87) *are given by*

$$\begin{aligned} M(x)\,\dot{v} + \dot{M}(x)\,v - \frac{1}{2}\, S^{-\mathsf{T}}(x)\, \frac{\partial^{\mathsf{T}}(M(x)\,v)}{\partial x}\, v + \\ S^{-\mathsf{T}}(x)\left(\dot{S}^{\mathsf{T}}(x) - \frac{1}{2}\frac{\partial^{\mathsf{T}} v}{\partial x}\right) M(x)\,v + S^{-\mathsf{T}}(x)\,\frac{\partial \mathcal{U}(x)}{\partial x} = \tau \end{aligned} \tag{2.88}$$

A sketch of the proof is presented in [40]. However, the constant 1/2 on the fourth term of the left-hand side of (2.88) is missing in [40]. This was caused by a differentiation error of a quadratic form in Equation (17) of [40]. A complete demonstration can be found in Section B.1.

Theorem 1 follows from Lemma 1 and provides the basis in which the equations of motion for a vehicle-manipulator system are derived.

Theorem 1. *The equations of motion given by* (2.88) *can be rewritten as follows*

$$M(x)\,\dot{v} + C(x, v)\,v + G(x) = \tau\,, \tag{2.89}$$

where

- $M(x) \in \mathbb{R}^{N\times N}$ *is a symmetric and positive-definite inertia matrix;*
- *The Coriolis matrix* $C(x,v) \in \mathbb{R}^{N\times N}$ *is given by*

$$C(x,v) = \sum_{k=1}^{N} \alpha_k(x)\, v_k + \sum_{k=1}^{N} \beta_k(x)\, v_k \tag{2.90}$$

with the components of matrices α_k, $\beta_k \in \mathbb{R}^{N\times N}$ *given by*

$$(\alpha_k)_{ij} = \sum_{l}^{N} \left(\frac{\partial M_{ij}}{\partial x_l}\, S^{-1}_{lk} - \frac{1}{2} S^{-1}_{li} \frac{\partial M_{jk}}{\partial x_l} \right) \tag{2.91}$$

$$(\beta_k)_{ij} = \sum_{l,m,s}^{N} \left(S^{-1}_{li} \left(\frac{\partial S_{sl}}{\partial x_m} - \frac{\partial S_{sm}}{\partial x_l} \right) S^{-1}_{mk} \right) M_{sj}\,; \tag{2.92}$$

- *The potential vector* $G(x) \in \mathbb{R}^N$ *is simply given by*

$$G(x) = S^{-\mathsf{T}}(x)\, \frac{\partial \mathcal{U}(x)}{\partial x}\,. \tag{2.93}$$

A sketch of the proof focused on the derivation of (2.91) and (2.92) is presented in [40]. It performs the component-wise expansion of the terms in (2.88). Although the final expressions for $(\alpha_k)_{ij}$ and $(\beta_k)_{ij}$ are correct in [40], some details on the presented demonstration need additional comments. First, there is a mistake in Eq. (32) of [40]: an additional term is missing because of the dependance of $v \in \mathbb{R}^N$ on $x \in \mathbb{R}^N$. However, it happens to be the same value as the term with 1/2 multiplication pointed out on the sketch of proof of Theorem 1, as a result of the previous mistake. These two terms sum up and the final expression for $(\beta_k)_{ij}$ is indeed correct in (37) of [40]. There is also a summation over k that should not exist in Eq. (36) of [40] and was corrected here, in (2.91). A correct proof of Theorem 1 is available in Section B.2.

Using the generalized coordinates of the vehicle-manipulator system $x = \xi_i = \begin{bmatrix} \eta_i^{\mathsf{T}} & q^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}} \in \mathbb{R}^{6+n}$ and the quasi-velocities $v = \zeta_i = \begin{bmatrix} (V_i^i)^{\mathsf{T}} & \dot{q}^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}} \in \mathbb{R}^{6+n}$ with respect to any frame $\mathbf{E}_i$, as defined in Section 2.1, they are clearly related through

$$\begin{bmatrix} V_i^i \\ \dot{q} \end{bmatrix} = \underbrace{\begin{bmatrix} J_i(\eta_{i_2}) & 0 \\ 0 & \mathsf{I}_3 \end{bmatrix}}_{S_i(\xi_i)} \begin{bmatrix} \dot{\eta}_i \\ \dot{q} \end{bmatrix}$$

$$\zeta_i = S_i(\xi_i)\, \dot{\xi}_i\,, \tag{2.94}$$

due to (2.27), which is equivalent to (2.83).

In the same way as in [19], the ISP is a vehicle-manipulator system with generalized coordinates $x = \xi_0$ and quasi-velocities $v = \zeta_0$ with respect to the vehicle frame $\mathbf{E}_0$ and related through (2.94), with $i = 0$. This way, Theorem 1 can be directly applied for the ISP model, and (2.89) becomes

$$\begin{bmatrix} M_{VV}^0 & M_{qV}^{0\,\mathsf{T}} \\ M_{qV}^0 & M_{qq}^0 \end{bmatrix} \dot{\zeta}_0 + \begin{bmatrix} C_{VV}^0 & C_{Vq}^0 \\ C_{qV}^0 & C_{qq}^0 \end{bmatrix} \zeta_0 + \begin{bmatrix} G_V^0 \\ G_q^0 \end{bmatrix} = \begin{bmatrix} F_0^0 \\ \tau_q \end{bmatrix},$$

$$M_0(\xi_0)\,\dot{\zeta}_0 + C_0(\xi_0, \zeta_0)\,\zeta_0 + G_0(\xi_0) = \tau_0\,. \tag{2.95}$$

where we have explicitly separated the vehicle and the manipulator dynamics. The generalized force acting on the system is $\tau^{\mathsf{T}} = \begin{bmatrix} F_0^{0\,\mathsf{T}} & \tau_q^{\mathsf{T}} \end{bmatrix} \in \mathbb{R}^{6+n}$, where $F_0^0 \in \mathbb{R}^6$ is the vector of generalized forces (wrenches) acting on the vehicle (collocated with V_0^0) and $\tau_q \in \mathbb{R}^n$ is the vector of generalized forces acting on the manipulator joints (collocated with $\dot{q} \in \mathbb{R}^n$).

Remark 9. *In* (2.95)*, it is clear that the vehicle motion affects the manipulator dynamics. The opposite is also true, however, the influence of the manipulator motion in the vehicle dynamics is expected to be small or even negligible in the case where the vehicle is much larger/massive than the manipulator.*

Thus, the lower part of (2.95) gives us the dynamic equation for the ISP:

$$M_{qq}^0\,\ddot{q} + C_{qq}^0\,\dot{q} + G_q^0 + \underbrace{M_{qV}^0\,\dot{V}_0^0 + C_{qV}^0\,V_0^0}_{\text{vehicle motion terms}} = \tau_q\,, \tag{2.96}$$

as previously seem in (1.4). Comparing (2.96) with the usual dynamics of robotic systems, the vehicle motion introduces two additional contributions on the joint torques, which depend on the vehicle velocities and accelerations. This is illustrated in Fig. 2.3. These torques can be compensated by a controller if V_0^0 and $\dot{V}_0^0$ are measured by inertial sensors or estimated, and suppose that the system parameters that define the matrices in (2.96) are known. It is worth noting that, while M_{qq} and M_{qV} depend only on q, C_{qq} and C_{qV} depend on the joint angles q and on the quasi-velocities ζ_0, and G_q depends on the generalized coordinates ξ_0.

Particularly, in the case of the ISP mounted on a vehicle, we do not have control over the vehicle motion, which is determined by unknown external wrenches $\tau_V \in \mathbb{R}^6$. These wrenches are determined by the vehicle thrusters and external environmental forces and torques, such as sea waves in the case of ships and wind, in the case of aerial vehicles. However, we do have control over the ISP motion, because its dynamics are determined by the joint torques $\tau_q \in \mathbb{R}^n$.

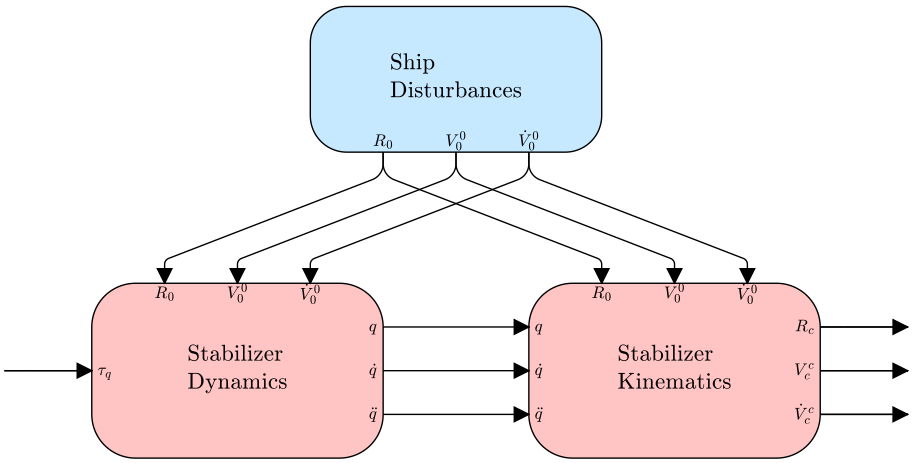


Figure 2.3: Block diagram of the ISP dynamic and kinematic models.

Using (2.94), we can also write the dynamic equation (2.95) in terms of the generalized coordinates ξ_0 and the time derivatives $\dot{\xi}_0$ and $\ddot{\xi}_0$ [19]

$$\widetilde{M}_0(\xi_0)\,\ddot{\xi}_0 + \widetilde{C}_0(\xi_0, \dot{\xi}_0)\,\dot{\xi}_0 + \widetilde{G}_0(\xi_0) = \widetilde{\tau}_0 \tag{2.97}$$

where

- $\widetilde{M}_0(\xi) \;=\; S_0^{\mathsf{T}}(\xi_0)\, M_0(q)\, S_0(\xi_0)$
- $\widetilde{C}_0(\xi_0, \dot{\xi}_0) \;=\; S_0^{\mathsf{T}}(\xi_0)\, \Big(C_0(\xi_0, \dot{\xi}_0)\, S_0(\xi_0) - M_0(q)\, S_0(\xi_0)\, \dot{S}_0^{-1}(\xi)\, S_0(\xi_0) \Big)$
- $\widetilde{G}_0(\xi_0) \;=\; S_0^{\mathsf{T}}(\xi_0)\, G_0(\xi_0)$
- $\widetilde{\tau}_0 \;=\; S_0^{\mathsf{T}}(\xi_0)\, \tau_0$, collocated with $\dot{\xi}_0$.

Remark 10. *Note that, since $T_0(\eta_{0_2})$ appears in the kinematics as a part of $S_0(\xi_0)$ in* (2.94)*, singularities can arise in* (2.97)*. In our case, using the RPY representation for orientation, we have singularities when $\theta_0 = \pm\pi/2$, that is, when the pitch angle is $\pm 90^\circ$ (nadir or zenith). Although singularities typically lead to computational problems and should receive special attention, for a wide range of applicatios, we don't need to worry about them because it is virtually impossible to achieve regions near the singularities (in the case where the vehicle is a ship, it would have to be pointed upwards or downwards).*

In a similar way, one could apply Theorem 1 to the same vehicle manipulator system, but with generalized coordinates defined as $x = \xi_c$ and quasi-velocities $v = \zeta_c$ with respect to the camera frame $\mathbf{E}_c$, also related through (2.94), with $i = c$. This alternate representation for the VMS states can be linked to the previous one

through the *quasi-velocity representation transformation*

$$\zeta_c = J_\zeta(q)\,\zeta_0\,, \qquad J_\zeta(q) = \begin{bmatrix} Ad_{g_{oc}}^{-1} & J_{0c}^c \\ 0 & \mathsf{I}_n \end{bmatrix}. \tag{2.98}$$

Transforming (2.95) according to (2.98), yields

$$M_c(\xi_c)\,\dot{\zeta}_c + C_c(\xi_c,\zeta_c)\,\zeta_c + G_c(\xi_c) = J_\zeta^{-\mathsf{T}}\,\tau_0\,. \tag{2.99}$$

Further simplifying (2.99), the dynamic equations get

$$\begin{bmatrix} M_{VV}^c & {M_{qV}^c}^{\mathsf{T}} \\ M_{qV}^c & M_{qq}^c \end{bmatrix}\dot{\zeta}_c + \begin{bmatrix} C_{VV}^c & C_{Vq}^c \\ C_{qV}^c & C_{qq}^c \end{bmatrix}\zeta_c + \begin{bmatrix} G_V^c \\ G_q^c \end{bmatrix} = \begin{bmatrix} F_c^c \\ \tau_q \end{bmatrix}, \tag{2.100}$$

where in this case, the generalized force acting on the VM system is $\tau_c^{\mathsf{T}} = \begin{bmatrix} {F_c^c}^{\mathsf{T}} & \tau_q^{\mathsf{T}} \end{bmatrix} \in \mathbb{R}^{6+n}$, where $F_c^c \in \mathbb{R}^6$ is the vector of generalized forces (wrenches) acting directly on the camera frame (collocated with V_c^c), related to F_0^0 by [21]

$$F_c^c = Ad_{0c}^{\mathsf{T}}\,F_0^0\,.$$

Remark 11. *Since $\tau_c \neq J_\zeta^{-\mathsf{T}}\,\tau_0$, the submatrices in* (2.100) *are not equal to the submatrices of $M_c(\xi_c)$, $C_c(\xi_c,\zeta_c)$ and $G_c(\xi_c)$ in* (2.99)*.*

The upper part of (2.99) gives us the dynamic equations of the camera itself:

$$M_{VV}^c\,\dot{V}_c^c + C_{VV}^c\,V_c^c + G_V^c + {M_{qV}^c}^{\mathsf{T}}\,\ddot{q} + C_{Vq}^c\,\dot{q} = F_c^c\,. \tag{2.101}$$

On the other hand, the lower part of (2.99) also gives us the dynamic equation for the ISP, but with the base motion variables given in terms of the camera motion, instead of the vehicle motion:

$$M_{qq}^c\,\ddot{q} + C_{qq}^c\,\dot{q} + G_q^c + M_{qV}^c\,\dot{V}_c^c + C_{qV}^c\,V_c^c = \tau_q\,. \tag{2.102}$$

Remark 12. *Equations* (2.95) *and* (2.99) *equally represent the dynamics of the VMS, but in opposite ways. Equation* (2.95) *considers the vehicle as the base of the VMS, while* (2.99) *considers the last link (camera) as the base of the VMS.*

It is worth mentioning that, in a similar way than in (2.61), (2.95) and (2.99) are also *linear* with respect to the *dynamic* parameters $\Pi_d \in \mathbb{R}^{N_d}$, which are composed by combinations of the links masses m_i, inertia tensor components and $p_{i,\bar{i}}^i \in \mathbb{R}^3$

components [20]:

$$Y_0(\xi_0, \zeta_0, g, \Pi_g)\,\Pi_d = \begin{bmatrix} Y_V^0 \\ Y_q^0 \end{bmatrix} \Pi_d = \tau_0\,, \tag{2.103}$$

$$Y_c(\xi_c, \zeta_c, g, \Pi_g)\,\Pi_d = \begin{bmatrix} Y_V^c \\ Y_q^c \end{bmatrix} \Pi_d = \tau_c\,. \tag{2.104}$$

The matrices $Y_0, Y_c \in \mathbb{R}^{(6+n)\times N_d}$ are known as *dynamic regressors*, and are fully configuration dependent. Linearity on the dynamic parameters is a key property for robust/adaptive control schemes, and for robustness analysis of control laws for mechanical systems as well.

Of course, the dynamic model of a 3 DOF ISP with revolution joints can be found by applying the methodology of Section 2.3.1 for the case of $n = 3$. However, note that non-conservative forces such as friction were not taken into account in the Lagrange model developed in the previous section. These forces might induce non-negligible effects in the ISP dynamics, and therefore, must be taken into account by the model. They are represented by an additional disturbance on the joint torques $\tau_f \in \mathbb{R}^n$ affecting the ISP dynamic equation.

Joint friction torques can be represented as the sum of Stribeck, Coulomb and viscous friction components [41]. The Stribeck friction F_s is the negatively sloped

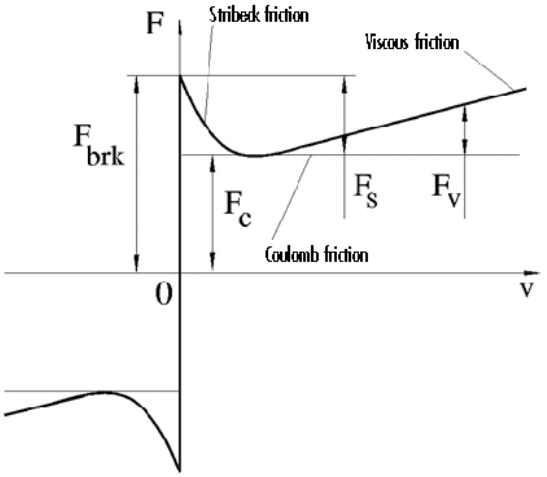


Figure 2.4: Simulated joint torque friction. Image from https://www.mathworks.com/help/physmod/simscape/ref/translationalfriction.html.

characteristics taking place at low velocities, as illustrated in Fig. 2.4. The Coulomb friction F_c results in a constant force at any velocity. The viscous friction F_v opposes motion with a force that is directly proportional to the joint velocity. The sum of the Coulomb and Stribeck frictions at the vicinity of zero velocity is often referred

to as the *breakaway* friction, F_{brk}. Therefore, an expression for the friction simulated in the i-th joint of the ISP is given by [42]

$$\tau_{f_i} = \sqrt{2\,e}\,(F_{brk_i} - F_{c_i})\,\exp\left(\frac{-x_{2_i}^2}{\omega_{s_i}^2}\right)\left(\frac{x_{2_i}}{\omega_{s_i}}\right) + F_{c_i}\,\tanh\left(\frac{x_{2_i}}{\omega_{c_i}}\right) + F_{v_i}\,x_{2_i}\,, \quad (2.105)$$

where $\omega_{s_i} = \sqrt{2}\,\omega_{brk}$ is the Stribeck velocity threshold, with ω_{brk_i} being the breakaway friction velocity. Here, ω_{c_i} is the Coulomb velocity threshold, and is generally considered to be 10 times lower than ω_{brk_i} [42].

Besides friction, cables passing through the internal ISP structure can store potential energy, leading to non-negligible spring-like effects [1]. Although elastic forces are conservative and thus can be derived from a potential function, they were not taken into account on the system Lagrangian function. Instead, they can be directly introduced into the joint torques as a linear function of the joint angles:

$$\tau_s = -K_s\,q\,, \qquad K_s > 0\,. \quad (2.106)$$

With these effects, the ISP equations of motion can be expressed by

$$M_{qq}^0\,\ddot{q} + C_{qq}^0\,\dot{q} + G_q^0 + M_{qV}^0\,\dot{V}_0^0 + C_{qV}^0\,V_0^0 + \tau_f + \tau_s = \tau_q\,, \quad (2.107)$$

$$M_{qq}^c\,\ddot{q} + C_{qq}^c\,\dot{q} + G_q^c + M_{qV}^c\,\dot{V}_c^c + C_{qV}^c\,V_c^c + \tau_f + \tau_s = \tau_q\,. \quad (2.108)$$

Next, for the sake of completeness, we develop explicit expressions for the matrices in (2.95), following the same methodology found in [19] to compute M_0, C_0 and G_0. The matrices M_c, C_c and G_c of (2.99) are not going to be computed explicitly, since they can be computed using numerical algorithms.

2.3.1.1 Computation of the Mass Matrix

The mass matrix $M_0(\xi_0)$ can be computed directly from (2.86), by simply deriving the total kinetic energy of the system.

The kinetic energy of each rigid body on the kinematic chain is given by

$$\mathcal{K}_i = \frac{1}{2}\,(V_i^i)^\intercal\,\bar{I}_i^i\,V_i^i\,. \quad (2.109)$$

Summing up for the $n+1$ bodies (the n links and the vehicle) and the vehicle and

using (2.51), the total kinetic energy of the system is

$$\mathcal{K} = \sum_{i=0}^{n} \mathcal{K}_i = \frac{1}{2}\, \zeta_0^{\mathsf{T}}\, M_0(\xi_0)\, \zeta_0\,, \tag{2.110}$$

where the VMS mass matrix $M_0(\xi_0) \in \mathbb{R}^{(6+n)\times(6+n)}$ as defined in Theorem 1 is

$$\begin{aligned} M_0(\xi_0) &= \begin{bmatrix} \bar{I}_0^0 & 0 \\ 0 & 0 \end{bmatrix} + \sum_{i=1}^{n} M_i^0(\xi_0)\,, \\ M_i^0(\xi_0) &= \begin{bmatrix} Ad_{g_{0i}}^{-\mathsf{T}}\, \bar{I}_i^i\, Ad_{g_{0i}}^{-1} & Ad_{g_{0i}}^{-\mathsf{T}}\, \bar{I}_i^i\, J_{0i}^i \\ (J_{0i}^i)^{\mathsf{T}}\, \bar{I}_i^i\, Ad_{g_{0i}}^{-1} & (J_{0i}^i)^{\mathsf{T}}\, \bar{I}_i^i\, J_{0i}^i \end{bmatrix}. \end{aligned} \tag{2.111}$$

Remark 13. *Since M_i^0 depends only on $q \in \mathbb{R}^n$, M_0 is actually a function of the joint angles $q \in \mathbb{R}^n$ only, and will be referred this way hereafter.*

Using (2.98), it is also possible to write the VMS kinetic energy with respect to the camera generalized velocities, as

$$\mathcal{K} = \sum_{i=0}^{n} \mathcal{K}_i = \frac{1}{2}\, \zeta_c^{\mathsf{T}}\, M_c(\xi_c)\, \zeta_c\,, \tag{2.112}$$

where the alternative VMS mass matrix $M_c(\xi_c) \in \mathbb{R}^{(6+n)\times(6+n)}$ is

$$\begin{aligned} M_c(\xi_c) &= \sum_{i=0}^{n-1} M_i^c(\xi_c) + \begin{bmatrix} Ad_{g_{nc}}^{\mathsf{T}}\, \bar{I}_n^n\, Ad_{g_{nc}} & 0 \\ 0 & 0 \end{bmatrix}, \\ M_i^c(\xi_c) &= \begin{bmatrix} Ad_{g_{ic}}^{\mathsf{T}}\, \bar{I}_i^i\, Ad_{g_{ic}} & -Ad_{g_{ic}}^{\mathsf{T}}\, \bar{I}_i^i\, Ad_{g_{ic}}\, J_{ic}^c \\ -(J_{ic}^c)^{\mathsf{T}}\, Ad_{g_{ic}}^{\mathsf{T}}\, \bar{I}_i^i\, Ad_{g_{ic}} & (J_{ic}^c)^{\mathsf{T}}\, Ad_{g_{ic}}^{\mathsf{T}}\, \bar{I}_i^i\, Ad_{g_{ic}}\, J_{ic}^c \end{bmatrix}, \end{aligned} \tag{2.113}$$

where the Jacobian matrix $J_{ic}^c \in \mathbb{R}^{6\times n}$ is defined by $V_{ic}^c = J_{ic}^c\, \dot{q}$.

As $M_0(q)$, $M_c(\xi_c)$ is also dependent on the joint angles only. Using the *representation transformation* in (2.98) and the expressions for the system kinetic energy in (2.110) and (2.112), the two mass matrices can be related through

$$M_0(q) = J_\zeta^{\mathsf{T}}(q)\, M_c(q)\, J_\zeta(q)\,. \tag{2.114}$$

It is also worth noticing that both $M_0(q)$ and $M_c(q)$ are symmetric and positive-definite, which is evident from the expressions of the link mass matrices M_i^0, M_i^c.

2.3.1.2 Computation of the Coriolis Matrix

The form of the Coriolis matrix is actually not unique [20]. The expression for (2.90) in the case of a vehicle-manipulator system in $SE(3)$ with quasi-velocities given by (2.94) can actually be further simplified using the following theorem.

Theorem 2. *The Coriolis matrix for a vehicle-manipulator system with quasi-velocities given by* (2.94)*, equations of motion given by* (2.89) *and whose vehicle is a free rigid-body in* $SE(3)$ *is*

$$C_0(q,\zeta_0) = \sum_{k=1}^{n} \frac{\partial M_0(q)}{\partial q_k}\,\dot{q}_k - \frac{1}{2}\bar{C}_0(q,\zeta_0)\,, \tag{2.115}$$

$$\bar{C}_0(q,\zeta_0) = \begin{bmatrix} -2\widetilde{ad}_{[\,M_{VV}^0 \;\; M_{qV}^{0\,\mathsf{T}}\,]\zeta_0} & 0 \\ \frac{\partial^{\mathsf{T}}}{\partial q}([\,M_{VV}^0 \;\; M_{qV}^{0\,\mathsf{T}}\,]\zeta_0) & \frac{\partial^{\mathsf{T}}}{\partial q}([\,M_{qV}^0 \;\; M_{qq}^0\,]\zeta_0) \end{bmatrix}.$$

For an arbitrary vector $p = \begin{bmatrix} p_1^{\mathsf{T}} & p_2^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}} \in \mathbb{R}^6$ *(*$p_1, p_2 \in \mathbb{R}^3$*), the operator* $\widetilde{ad}_p$ *is*

$$\widetilde{ad}_p = -\begin{bmatrix} \mathsf{0}_3 & \mathbf{S}(p_1) \\ \mathbf{S}(p_1) & \mathbf{S}(p_2) \end{bmatrix} \in \mathbb{R}^{6\times 6}\,.$$

A proof can be found in [19, Section 8.3.3.2]. It uses (2.90), (2.91), (2.92) and the structural properties of $M_0(q)$ and $S(\xi_0)$ to expand (2.90) into (2.115). Note that, in order to make the final expression of the Coriolis matrix consistent with the definitions in [19, Section 6.4.3.2], we have corrected the signal of the $\widetilde{ad}_{[\,M_{VV}^0 \;\; M_{qV}^{0\,\mathsf{T}}\,]\zeta_0}$ term in $\bar{C}_0$ of (2.115).

2.3.1.3 Computation of the Potential Vector

In the case of gravitational energy only, the potential vector (2.93) for (2.95) can be rewritten using an approach based on *energy conservation*. Note that $\tau_{\bar{g}} = G_0(\xi_0)$ is the generalized force vector needed to cancel out $G_0(\xi_0)$ in (2.89). The infinitesimal work done by $\tau_{\bar{g}}$ is

$$dW_{\bar{g}} = \tau_{\bar{g}}^{\mathsf{T}}\,\zeta_0\,dt\,.$$

The generalized gravitational force acting on body i is

$$F_g^i = -m_i\,g\begin{bmatrix} R_i^{\mathsf{T}} z_0 \\ \mathbf{S}(p_{i\bar{i}}^i)\,R_i^{\mathsf{T}} z_0 \end{bmatrix} \in \mathbb{R}^6\,, \tag{2.116}$$

where g is the gravitational acceleration constant. The infinitesimal work done by

the gravitational forces is given by

$$dW_g = \sum_{i=0}^{n} (F_g^i)^\mathsf{T} V_i^i \, dt \, .$$

Then, if only gravitational forces are present, by conservation of energy:

$$dW_{\bar{g}} + dW_g = 0 \, . \tag{2.117}$$

Finally, expanding (2.117) and comparing its terms, we can use (2.51) to compute $G(\xi)$ as

$$G_0(\xi_0) = -\sum_{i=0}^{n} \begin{bmatrix} Ad_{g_{0i}}^{-1} & J_{0i}^i \end{bmatrix}^\mathsf{T} F_i^i \, . \tag{2.118}$$

where here J_{00}^0 is zero and $Ad_{g_{00}}^{-1} = \mathsf{I}_6$, by definition. It is worth commenting that a similar approach to compute (2.118) is used in [19], but the final expression obtained for $G_0(\xi_0)$ in [19] is incorrect.

Of course, this method for computing $G_0(\xi_0)$ in the case of a gravitational potential is equivalent to (2.93), but simpler. If other conservative forces need to be taken into account, such as spring-like effects, $G_0(\xi_0)$ must be solved directly by using (2.93).

2.3.2 Newton-Euler Algorithm

The *Newton Euler* method is a computationally efficient algorithm that can be used to numerically solve the *inverse dynamics* problem for the VMS dynamic equation (2.89). Given $\xi_j, \zeta_j, \dot{\zeta}_j \in \mathbb{R}^{6+n}$ associated to an arbitrary frame $\mathbf{E}_j$ rigidly attached to the i-th body of the kinematic chain, $g \in \mathbb{R}$ and the all system parameters, the *inverse dynamics* problem consists in computing the forces/torques (or wrenches) needed to perform a given input motion in the VM system. On its most general form, it can be expressed algebraically by

$$\tau_j = \mathsf{InvDyn}(\xi_j, \zeta_j, \dot{\zeta}_j, g, \Pi) \, , \tag{2.119}$$

where $\tau_j^\mathsf{T} = \begin{bmatrix} {F_j^j}^\mathsf{T} & \tau_q^\mathsf{T} \end{bmatrix} \in \mathbb{R}^{6+n}$, with $F_j^j \in \mathbb{R}^6$ being the vector of generalized forces (wrenches) acting directly on $\mathbf{E}_j$ (collocated with V_j^j). The parameter vector $\Pi^\mathsf{T} = \begin{bmatrix} \Pi_g^\mathsf{T} & \Pi_d^\mathsf{T} \end{bmatrix}$ contains combinations of the geometric and dynamic parameters.

The Newton-Euler algorithm is composed of two steps. The first one is the *propagation of velocities and accelerations* upwards the kinematic chain, summarized in Algorithm 2. The second one consists in solving the dynamic equations of motion

for each rigid body in the system, starting from the n-th link and ending up on the vehicle frame $\mathbf{E}_0$.

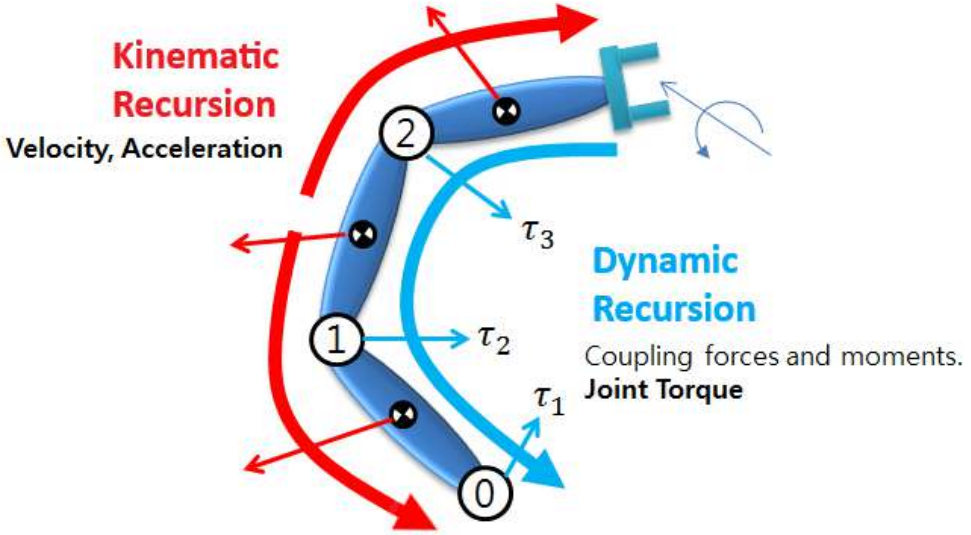


Figure 2.5: Illustration of the two steps of the Newton-Euler algorithm for a serial robot. Edited from http://www.matlabinuse.com/Mastering_MATLAB/11349.

Algorithm 5 (Backward Propagation of Wrenches)**.** *The Newton-Euler equations for the contact body wrenches $F_k^k \in \mathbb{R}^6$ between the bodies of the vehicle-manipulator system (links and vehicle) are*

$$F_{k-1}^{k-1} = \Phi_{k,k-1}^{\mathsf{T}}\, \Omega_{k-1,k}\, F_k^k + M_{k-1}\, \dot{V}_{k-1}^{k-1} + B_{k-1}\,, \tag{2.120}$$

$$M_k = \begin{bmatrix} m_k\, \mathbf{I}_3 & -m_k\, \mathbf{S}(p_{k\bar{k}}^k) \\ m_k\, \mathbf{S}(p_{k\bar{k}}^k) & I_k^k \end{bmatrix},$$

$$B_k = \begin{bmatrix} m_k\, \mathbf{S}(\omega_k^k)\, (\mathbf{S}(\omega_k^k)\, p_{k\bar{k}}^k + v_k^k) \\ m_k\, \mathbf{S}(p_{k\bar{k}}^k)\, \mathbf{S}(\omega_k^k)\, v_k^k + \mathbf{S}(\omega_k^k)\, I_k^k\, \omega_k^k \end{bmatrix},$$

where the parameters $p_{k\bar{k}}^k$, m_k and I_k^k compose Π_d in (2.119)*. These equations must be solved from $k = n = 3$ to $k = 0$, using the velocity and acceleration twists V_k^k and $\dot{V}_k^k$ previously computed in Algorithm 2. Also, we set here $n + 1 = c$ and we do not consider external wrenches acting on the camera frame $\mathbf{E}_c$, so that $F_{n+1}^{n+1} = 0$.*

Algorithm 6 (Newton-Euler Algorithm)**.** *Let $\mathbf{E}_j$ be any frame rigidly attached to the i-th body of the kinematic chain, with known transformation $g_{ij} \in SE(3)$. Supposing known geometric and dynamic parameters, pose $g_j \in SE(3)$, velocities V_j^j and accelerations $\dot{V}_j^j$ of $\mathbf{E}_j$ and joint angles q, velocities $\dot{q}$ and accelerations $\ddot{q}$, the Newton-Euler algorithm is described by the steps below:*

1. *Compute V_i^i and $\dot{V}_i^i$ associated to the i-th body frame $\mathbf{E}_i$, using g_{ij},* (2.16) *and* (2.22)*;*

2. *Compute all velocity and acceleration twists V_k^k and $\dot{V}_k^k$ associated to each link of the kinematic chain using Algorithm 2;*

3. *Compute all body contact wrenches F_k^k associated to each link of the kinematic chain from $k = n$ to $k = 0$ using Algorithm 5. The effect of gravity (in $-z_0$ direction) can be introduced by modifying $\dot{V}_k^k$ for each link according to:*

$$\dot{V}_k^k \leftarrow \dot{V}_k^k - g \begin{bmatrix} R_k^\mathsf{T} z_0 \\ 0 \end{bmatrix}, \tag{2.121}$$

   where $g \in \mathbb{R}$ is the gravitational acceleration on Earth;

4. *Compute the joint torques by projecting the wrenches acting on frames $\mathbf{E}_k$ into their rotation axis:*

$$\tau_{q_k} = H_k^\mathsf{T} \, \Omega_{k-1,k} \, F_k^k \,. \qquad k = 1, ..., n \,. \tag{2.122}$$

The wrenches $F_j^j \in \mathbb{R}^6$ acting on frame $\mathbf{E}_j$ fixed to the i-th rigid-body are simply

$$F_j^j = Ad_{g_{ij}}^\mathsf{T} \, F_i^i \in \mathbb{R}^6 \,.$$

Note that (2.122) on Algorithm 6 can be used to compute the joint torques τ_q in (2.96) or (2.102) needed to perform a given motion on the ISP. If $\mathbf{E}_j \equiv \mathbf{E}_0$, then these torques are equivalent to (2.96), while if $\mathbf{E}_j \equiv \mathbf{E}_c$, they are equivalent to (2.102).

Summarizing, the Newton-Euler algorithm can be used to solve the inverse dynamics problem for both (2.95) and (2.99) just by initializing the algorithm with the appropriate inputs. This can be seem by rewriting (2.119) as

$$\tau_0 = \mathsf{InvDyn}(\xi_0, \zeta_0, \dot{\zeta}_0, g, \Pi) \,, \tag{2.123}$$

$$\tau_c = \mathsf{InvDyn}(\xi_c, \zeta_c, \dot{\zeta}_c, g, \Pi) \,, \tag{2.124}$$

from where the relation with (2.95) and (2.99) can be easily seem.

Algorithm 6 can also be used to compute matrices and terms of (2.95) and (2.99) separately.

Algorithm 7 (Computation of the Mass Matrices)**.** *The j-th column of the system mass matrix $M_0(q) \in \mathbb{R}^{(6+n)\times(6+n)}$ in* (2.95) *is equal to $\tau_j^j \in \mathbb{R}^{6+n}$, the generalized*

force obtained by executing Algorithm 6 with $\zeta_0 = 0$, $g = 0$ *and* $\dot{\zeta}_0$ *equal to a generalized acceleration whose* j*-th component is* 1 *and all remaining components are* 0*. Therefore,* $M_0(q)$ *can be fully computed (column by column) by executing Algorithm 6* $6 + n$ *times, where* n *is the number of ISP joints.*

Algorithm 8 (Computation of the Coriolis Term)**.** *The Coriolis term* $C_0(\xi_0, \zeta_0)\,\zeta_0 \in \mathbb{R}^{6+n}$ *in* (2.95) *can be obtained by executing Algorithm 6 with* $\dot{\zeta}_0 = 0$ *and* $g = 0$*.*

Algorithm 9 (Computation of the Gravity Vector)**.** *The gravity vector* $G_0(\xi_0) \in \mathbb{R}^{6+n}$ *in* (2.95) *can be obtained by executing Algorithm 6 with* $\zeta_0 = \dot{\zeta}_0 = 0$*.*

A similar version of these algorithms can be used to compute the terms of (2.99) as well, just by replacing ξ_0, ζ_0, $\dot{\zeta}_0$, $M_0(q)$, C_0, G_0 by ξ_c, ζ_c, $\dot{\zeta}_c$, $M_c(q)$, C_c and G_c, respectively.

Using Algorithms 7, 8 and 9 and separating the matrices and terms the same way as in (2.95), the mass matrices M^0_{qq}, M^0_{qV}, the gravity vector G^0_q and the Coriolis term $C^0_{qq}\,\dot{q} + C^0_{qV}\,V^0_0$ for the ISP dynamics can be extracted. The same is valid for matrices M^c_{qq}, M^c_{qV}, gravity vector G^c_q and Coriolis term $C^c_{qq}\,\dot{q} + C^c_{qV}\,V^c_c$. This is important because, in our future developments, we will not be interested in the base dynamics; only in the ISP dynamics, given by (2.96) or (2.102).

Finally, the Newton-Euler algorithm can also solve the *forward dynamics* problem of (2.95). It consists in the computation of $\dot{\zeta}_0 \in \mathbb{R}^{6+n}$, given the system complete states of motion $\xi_0, \zeta_0 \in \mathbb{R}^{6+n}$ and the system generalized forces (or wrenches) $\tau_0 \in \mathbb{R}^{6+n}$. Algorithmically, the forward dynamics problem can be expressed by

$$\dot{\zeta}_0 = \mathsf{ForDyn}(\tau_0, \xi_0, \zeta_0, g, \Pi)\,. \tag{2.125}$$

Algebraically, the solution is computed by solving (2.95) for $\dot{\zeta}_0$:

$$\dot{\zeta}_0 = M_0^{-1}\,(\tau_0 - \tau_0^*)\;, \qquad \tau_0^* = C_0\,\zeta_0 + G_0\,. \tag{2.126}$$

This solution always exists, since the mass matrix $M_0(q)$ is of full rank. To compute it numerically, note that Algorithm 7 can be used to compute $M_0(q)$ and τ_0^* is simply the Coriolis term summed to the gravity vector, which can be computed using Algorithms 8 and 9. Therefore, given an input generalized force $\tau_0 \in \mathbb{R}^{6+n}$, $\dot{\zeta}_0$ can be computed by (2.126).

Integrating the generalized *accelerations* $\dot{\zeta}_0$ twice and using the inverse of (2.94), one can compute the current system states ξ_0 and ζ_0, for a given initial condition. This is why the forward dynamics problem can be used for the simulation of the vehicle-manipulator system [43].

Chapter 3

P-PI and Computed Torque Control

In this section, we formulate the control laws of two different approaches for LOS stabilization: the commonly used P-PI control and our proposed Computed Torque-PID (CT-PID) control. These techniques could be referred to as *classic* techniques, since despite of their differences, both are profoundly based in linear control.

As previously mentioned, the P-PI control law is widely used in literature for LOS stabilization. It is a simple linear controller that provides a satisfactory performance for most stabilization applications, which explains why it is still being employed in real applications despite of the existence of more involved control techniques. On the other hand, we have CT-PID control, which is a feedback-linearization-based technique, used mostly in robotics. It takes advantage of the complex dynamic model developed in the last chapter, and could (in principle) provide better performance for the LOS stabilization problem, provided that the ISP model is well known.

A fundamental assumption carried out not only in this chapter but in this entire work is *torque* control. It assumes that the ISP joint torques can be directly controlled by means of a high-performance motor driver, which is a reasonable assumption due to the availability of such drivers in industry nowadays. We also assume that the torque delivered by the motor axis is the same as the torque delivered in the joint axis; which is also reasonable in the case of ISPs, which are generally driven by *direct drive* motors, as explained in Section 1.1.

Torque control can be expressed analytically as

$$u(t) := \tau_q \,. \tag{3.1}$$

where $u(t) \in \mathbb{R}^3$ is the system control variable.

3.1 P-PI Control

The P-PI controller is widely used in LOS stabilization due to its simplicity, ease of tuning and acceptable performance for most applications. To obtain tuned P-PI gains according to a reasonable criteria, we develop an analogy between the dynamic model in (2.96) with the simpler dynamic model considered in several works, as [1, 2, 14].

From (2.102), let $M^c_{qV} = \begin{bmatrix} M^c_{qV_1} & M^c_{qV_2} \end{bmatrix}$, with $M^c_{qV_1}, M^c_{qV_2} \in \mathbb{R}^{3\times 3}$. Using the representation Jacobian in (2.24), (2.102) can be rewritten as

$$M_\eta \, \ddot{\eta}_{c_2} + \tau^c_d = \tau_q \,, \tag{3.2}$$

where $M_\eta = M^c_{qV_2} \, T_c$ is an operational space mass matrix and $\tau^c_d \in \mathbb{R}^3$ is a disturbance given by

$$\tau^c_d = M^c_{qq}\ddot{q} + C^c_{qq}\dot{q} + G^c_q + C^c_{qV}V^c_c + M^c_{qV_1}\dot{v}^c_c - M_\eta \, \dot{T}_c^{-1}\omega^c_c \,, \tag{3.3}$$

where, for the first time, (2.30) was used in the last term of (3.3). Consider that M_η in (3.2) can be written as $M_\eta = \overline{M}_\eta + \Delta M_\eta(q)$, where $\overline{M}_\eta$ denotes the *constant* and *antidiagonal* part of M_η. In the ISP configuration, we use q to act in YPR (not RPY) in home position, hence the antidiagonal and not the diagonal part is taken. Then, we can rewrite (3.2) as

$$\overline{M}_\eta \, \ddot{\eta}_{c_2} + \overline{\tau}^c_d = \tau_q \,, \tag{3.4}$$

where $\overline{\tau}^c_d = \tau^c_d + \Delta M_\eta \, \ddot{\eta}_{c_2}$. Equation (3.4) is the simple dynamic model usually considered in the literature: a double-integrator with an inertia gain for each decoupled orientation DOF with a disturbance $\overline{\tau}_d$ to be attenuated by the control.

Define the RPY error as

$$e_\eta = \eta_{d_2} - \eta_{c_2} \,, \tag{3.5}$$

where $\eta_{d_2} \in \mathbb{R}^3$ is a RPY reference, whose expression in derived in Section 2.2.4. The P-PI control law can be explicitly written as:

$$z(t) = K_{p_1} \, e_\eta - \dot{\eta}_{c_2} \,, \tag{3.6}$$

$$u(t) = K_{p_2} \, z(t) + K_I \int_0^t z(t) \, dt \,, \tag{3.7}$$

where K_{p_1}, K_{p_2} and K_I are $\mathbb{R}^{3\times 3}$ diagonal matrices. Let $\overline{M}_{\eta_i}$ be the i-th antidiagonal

element of $\overline{M}_\eta \in \mathbb{R}^{3\times3}$. With (3.6) and (3.7) in (3.4), the characteristic polynomial of the closed loop system for the i-th orientation DOF is

$$\overline{M}_{\eta_i}\,\lambda^3 + K_{P_{2i}}\,\lambda^2 + (K_{P_{1i}}\,K_{P_{2i}} + K_{I_i})\,\lambda + K_{P_{1i}}\,K_{I_i} = 0\,. \tag{3.8}$$

For each of the $i = 1, 2, 3$ equations in (3.4), the gain tuning procedure consists in choosing one of the roots in (3.8) to be stable and sufficiently distant from the origin. The other two are chosen to match the poles of a second order characteristic polynomial to meet a given transient or bandwidth requirement.

Figure 3.1 illustrates the block diagram of the closed-loop system with the P-PI controller.

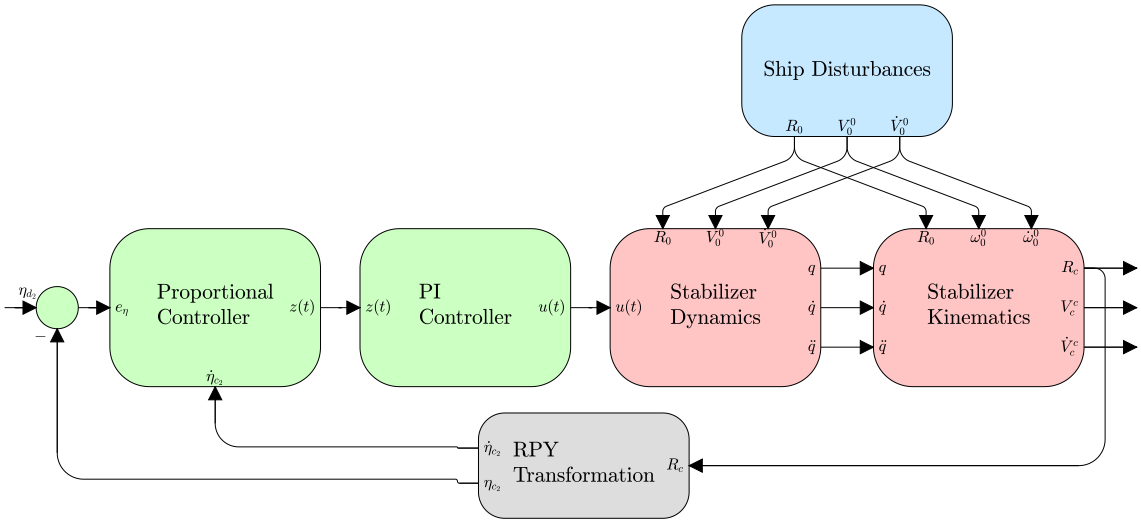


Figure 3.1: Block diagram for the closed-loop system with P-PI controller.

Remark 14. *This controller requires careful gain adjustment to provide a satisfactory performance. Even so, if the magnitude of the state-dependent disturbance $\overline{\tau}_d^c$is excessive, it may be impossible to achieve the required level of performance, specially for high accuracy applications.*

3.2 Computed-Torque Control

Computed torque control is usually used in robotics for joint or operational space control. It takes full advantage of the ISP dynamic model and is a more sophisticated alternative to the P-PI and other simple linear control methods. The basic idea is to cancel the system nonlinear disturbances using feedback, linearizing the system with an inner control loop. Since, in practice, *perfect* knowledge of system parameters is impossible, a state dependent disturbance will inevitably appear in the

dynamics. However, if the nominal and real system parameters are close enough, the stabilization/tracking performance of the system can still meets the requirements.

Before introducing the two computed torque schemes proposed for the stabilization and tracking problem, we present an useful result that will be used to demonstrate the stability of these controllers. It is a generalized version of the *Small Gain Theorem* [8] that gives the conditions for the BIBO stability of a feedback connection between two BIBO stable systems.

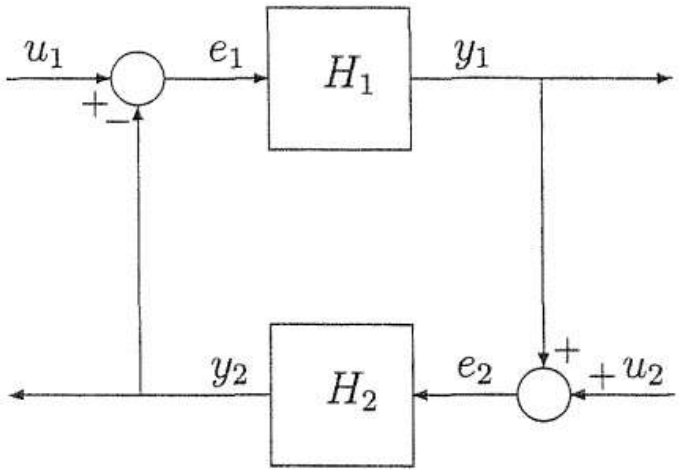


Figure 3.2: Feedback connection between two systems H_1 and H_2. Image from [8].

Theorem 3 (Generalized Small Gain Theorem)**.** *Consider two BIBO stable systems H_1 and H_2 from Fig. 3.2, satisfying the following BIBO stability conditions:*

$$\|y_1\| \leq \gamma \|e_1\| + \beta\,, \tag{3.9}$$

$$\|y_2\| \leq A \|e_2\|^2 + B \|e_2\| + C\,, \tag{3.10}$$

Suppose further that the feedback system is well defined in the sense that for every pair of inputs u_1 and u_2, there exist unique outputs y_1 and y_2. Define the input, output and error of the feedback system as

$$u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \; y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}, \; e = \begin{bmatrix} e_1 \\ e_2 \end{bmatrix}, \tag{3.11}$$

and

$$D_1 = \|u_1\| + A \|u_2\|^2 + (2A\beta + B) \|u_2\| + A\beta^2 + B\beta + C\,, \tag{3.12}$$

$$D_2 = \|u_2\| + \gamma \|u_1\| + \gamma C + \beta\,. \tag{3.13}$$

Then, under the following assumptions

(i) $\gamma B + 2\gamma A\,(\beta + \|u_2\|) < 1\,,$

(ii) $4\gamma^2 A\,D_1 < (1 - 2\gamma A(\|u_2\| + \beta) - \gamma B)^2\,,$

(iii) $4\gamma A\,D_2 < (1 - \gamma B)^2\,,$

there exist bounds ϵ_1 *and* ϵ_2 *such that, if* $\|e_1(0)\| < \epsilon_1$ *and* $\|e_2(0)\| < \epsilon_2$ *for* $t = 0$*, the feedback connection of Fig. 3.2 is BIBO stable.*

The demonstration is presented in Section B.3. Next, we present a corollary that links Theorem 3 to the usual small gain theorem, as considered in [8].

Corollary 3.1. *Considering* $A = 0$ *in Theorem 3, the stability conditions reduce to the usual form of the small gain theorem [8]:*

$$\gamma B < 1\,.$$

Proof. The proof is straightforward, since for $A = 0$, the BIBO stability conditions of Theorem 3 reduce to

(i) $\gamma B < 1\,,$

(ii) $0 < (1 - \gamma B)^2\,.$

The second condition is always satisfied. Then, the condition for the BIBO stability of the feedback connection is simply $\gamma B < 1$. □

3.2.1 Direct Computed-Torque Control

The first proposed computed torque scheme considers that the inertial sensors are placed in the direct configuration, allowing the camera disturbances to be directly measured. It tries to cancel the nonlinear disturbance τ_d^c in (3.2) directly, which can be estimated since the model is known and the camera pose, velocities and accelerations are directly measured.

The direct CT-PID control law can be written as:

$$z(t) = \ddot{\eta}_{d_2} + K_P\,e_\eta + K_I \int_0^t e_\eta(t)\,dt + K_D\,\dot{e}_\eta\,, \tag{3.14}$$

$$u(t) = \widehat{M}_\eta\,z(t) + \widehat{\tau}_d^c\,, \tag{3.15}$$

where K_P, K_I, $K_D \in \mathbb{R}^{3\times3}$ are diagonal gain matrices and $\widehat{M}_\eta$, $\widehat{\tau}_d^c$ in (3.2) are computed by the methods described in Section 2.3.2, but using *nominal* parameters $\widehat{\Pi}_g$ and $\widehat{\Pi}_d$.

Figure 3.3 illustrates the block diagram of the closed-loop system with the proposed CT-PID controller.

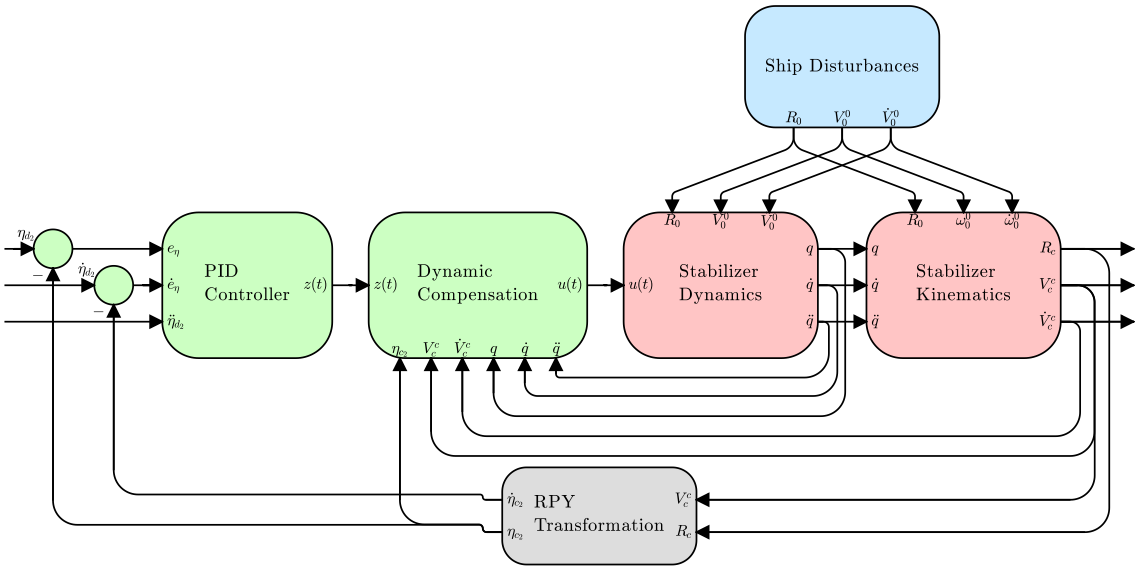


Figure 3.3: Block diagram for the closed-loop system with CT-PID controller.

The following theorem states the boundedness property of the RPY orientation error under the proposed control scheme and some basic assumptions.

Theorem 4. *Consider the plant dynamics is given by* (3.2)*. The error on the dynamic parameters is defined as* $\widetilde{\Pi}_d = \Pi_d - \widehat{\Pi}_d$*, where* Π_d *and* $\widehat{\Pi}_d$ *are real and nominal dynamic parameters, respectively. If the following assumptions hold:*

(i) $T_c(\eta_{c_2})$ *is non-singular for all* $t > 0$*,*

(ii) the geometric parameters Π_g *are known,*

(iii) the camera velocity and acceleration twists V_c^c*,* $\dot{V}_c^c$ *are norm-bounded,*

then, the direct CT-PID control laws (3.14) *and* (3.15) *ensure that there exists an attraction domain for* e_η *for some set of positive gains* K_P*,* K_D *and* K_I *and a limited parametric error norm* $\left\|\widetilde{\Pi}_d\right\|$*, and that* e_η *ultimately converges to a residual set* $\{\|e_\eta(t)\| < \rho \mid t \to \infty\}$ *of order* $\mathcal{O}(\left\|\widetilde{\Pi}_d\right\|)$*, i.e., the constant* ρ *is bounded by* $\left\|\widetilde{\Pi}_d\right\|$*. Moreover,* $\rho \to 0$ *as* $\left\|\widetilde{\Pi}_d\right\| \to 0$*.*

The proof is carried out in Section B.4 and is a direct application of Theorem 3.

Remark 15. *A clear disadvantage of this approach is the need for the first and second time derivatives of the joint angles* $q \in \mathbb{R}^3$ *in the computation of the compensating disturbance vector* τ_d^c *in* (3.15)*. Usually, estimators for the time-derivatives of practical signals suffer from several problems, such as high-frequency noises and*

delays that can compromise the estimation. If these estimated time-derivatives are used in control schemes, the control performance could be severely affected.

Consider the CT-PID control law (3.14), (3.15) with system (3.2). The closed-loop characteristic equation for the RPY error, considering perfect canceling of terms, is

$$\lambda^3 + K_{D_i}\,\lambda^2 + K_{P_i}\,\lambda + K_{I_i} = 0\,. \tag{3.16}$$

where K_{P_i}, K_{D_i}, K_{I_i}, $i = 1, 2, 3$ are the diagonal components of the PID gain matrices. Therefore, considering full knowledge of the system dynamic and geometric parameters, pole placement is a simple tuning method for this controller; appropriate PID gains can be chosen such that (3.16) represents the desired third-order dynamics.

3.2.2 Indirect Computed-Torque Control

Here, an alternative topology is proposed to control the camera LOS. As before, an inner controller implements a *computed torque* method for stabilization, but in the joint space. That means that the motion of the vehicle must be measured, ideally canceling the dynamic disturbances in (2.96), linearizing the system into a decoupled double integrator in the ISP *joint angles*. The outer controller is a *PID* in operational space with reference acceleration *feed-forward* and a linearization term to cancel further *kinematic* disturbances acting on the system. As before, the integral term attempts to reject the remaining disturbances due to imperfect knowledge of the system parameters, both geometric and dynamic.

Let the VMS model be described by (2.96), rewritten here as

$$M^0_{qq}\,\ddot{q} + \tau^0_d = \tau_q\,, \tag{3.17}$$

where τ_d is a disturbance given by

$$\tau^0_d = C^0_{qq}\,\dot{q} + G^0_q + M^0_{qV}\,\dot{V}^0_0 + C^0_{qV}\,V^0_0\,. \tag{3.18}$$

The *inner stabilization controller* is defined as

$$\begin{aligned} u(t) &= \widehat{M}^0_{qq}\,z(t) + \widehat{\tau}^0_d\,, \\ \widehat{\tau}^0_d &= \widehat{C}^0_{qq}\,\dot{q} + \widehat{G}^0_q + \widehat{M}^0_{qV}\,\dot{V}^0_0 + \widehat{C}^0_{qV}\,V^0_0\,, \end{aligned} \tag{3.19}$$

with $\widehat{M}^0_{qq} = M^0_{qq}(q, \widehat{\Pi}_g, \widehat{\Pi}_d)$, $\widehat{\tau}^0_d = \tau^0_d(\xi_0, \zeta_0, \dot{V}^0_0, \widehat{\Pi}_g, \widehat{\Pi}_d)$, where the "hat" operator stands for *estimated* parameter vector. In the hypothesis of known system param-

eters ($\widehat{\Pi}_g = \Pi_g$ and $\widehat{\Pi}_d = \Pi_d$), this controller ideally linearizes the system into a double integrator in joint space.

With RPY error defined the same way as (3.5), the *outer tracking controller* is

$$\lambda(t) = \ddot{\eta}_{d_2} + K_P\, e_\eta + K_I \int_0^t e_\eta(\tau) d\tau + K_D\, \dot{e}_\eta\,, \tag{3.20}$$

$$z(t) = (\widehat{J}_q)^{-1} \left(\lambda(t) - \widehat{L}_q\, \dot{q} - \widehat{J}_\omega\, \dot{\omega}_0^0 - \widehat{L}_\omega\, \omega_0^0 \right)\,, \tag{3.21}$$

where K_P, K_I, $K_D \in \mathbb{R}^{3\times 3}$ are gain matrices and $\widehat{J}_q = J_q(\widehat{\Pi}_g)$, $\widehat{L}_q = L_q(\widehat{\Pi}_g)$, $\widehat{J}_\omega = J_\omega(\widehat{\Pi}_g)$ and $\widehat{L}_\omega = L_\omega(\widehat{\Pi}_g)$. The remaining dependencies of matrices J_q, J_ω, L_q and L_ω were omitted for simplicity, but can be obtained from (2.60). Figure 3.4 illustrates a block diagram for the proposed control scheme.

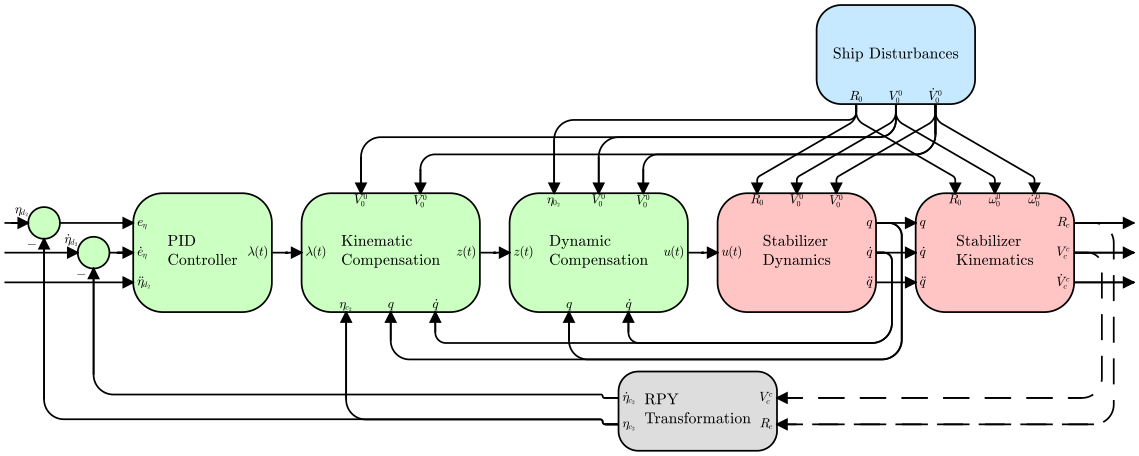


Figure 3.4: Block diagram for the closed-loop system with joint space CT-PID controller. Notice how the controller does not depend on $\ddot{q}$ in this configuration.

The following theorem shows that control laws (3.19), (3.20) and (3.21) lead to a stronger robustness property then control laws (3.14), (3.15) of Section 3.2.1, since it allows Assumption (ii) to be removed from Theorem 4.

Theorem 5. *Let* (2.96) *and* (2.60) *describe the system dynamics and kinematics. The error on the system parameters is defined as* $\widetilde{\Pi} = \Pi - \widehat{\Pi}$*, where* Π *and* $\widehat{\Pi}$ *are real and nominal parameters of the ISP, respectively. If the following assumptions hold:*

(i) $T_c(\eta_{c_2})$ *is non-singular for all* $t > 0$*,*

(ii) there are ϵ_1, $\epsilon_2 > 0$ *such that:* $\|V_0^0\| \leq \epsilon_1$, $\left\|\dot{V}_0^0\right\| \leq \epsilon_2$*,*

then, the indirect CT-PID control laws (3.19), (3.20) *and* (3.21) *ensure that there exists an attraction domain for* e_η *for some set of positive gains* K_P*,* K_D *and* K_I *and a limited parametric error norm* $\left\|\widetilde{\Pi}\right\|$*, and that* e_η *ultimately converges to a residual*

set $\{\|e_\eta(t)\| < \rho \mid t \to \infty\}$ *of order* $\mathcal{O}(\left\|\widetilde{\Pi}\right\|)$*, i.e., the constant* ρ *is bounded by* $\left\|\widetilde{\Pi}\right\|$*. Moreover,* $\rho \to 0$ *as* $\left\|\widetilde{\Pi}\right\| \to 0$*.*

The proof is carried out in Section B.5 and is also an application of Theorem 3.

The main practical advantage of this approach in comparison with the previous one is the absence of the joint accelerations in the control law. Besides, the stability analysis is also easier and takes the uncertainty in the *geometric parameters* in account as well, which was not considered in Theorem 4.

The tuning procedure for this controller is essentially the same as the one described in Section 3.2.1. Substituting control laws (3.19) and (3.21) onto (2.96) and (2.60) yields the same closed-loop RPY error characteristic equation as (3.16), on the assumption of perfect nonlinear canceling. Therefore, pole placement can once again be employed to match the desired third-order dynamics.

3.3 Simulation Results

This section compares the performance of the P-PI and the CT-PID methods for the LOS control problem. MATLAB Simulink® models were implemented for the simulation of the dynamic model of a 3-DOF ISP installed on a vessel, developed in Chapter 2. Figure 3.5 shows our proposed Matlab simulator, used throughout this work. The Simulink block "ISP System" implements all algorithms for the system simulation, while block "Controller" can selected among the proposed control strategies by means of an initialization script. "CG Ship Motion" contains Matlab *lookup tables* for the simulation of the vessel motion data provided by Tecgraf. The "Target Motion" block allows for the selection of the desired motion reference, while the "Sensors" block implements sensor hardware effects, such as such as resolution, bias, and noise. The effect of the sampling time on the controllers and sensor data acquisition was not taken into account. However, since most digital computers work at high sample rates, this effect should be negligible in a real world application. Encoders measure the joints positions q, with which it is possible to *estimate* $\dot{q}$ and $\ddot{q}$. Motor/driver electromechanical dynamics was not taken into account, but a saturation of $\pm 12.2\, Nm$ in each joint motor was considered.

The vessel motion data is represented by the vehicle variables η_0, V_0^0 and $\dot{V}_0^0$, and nine different sets of $1000\, s$ of data are available in the form of Matlab "*.mat*" files. These data were obtained from the simulations of FPSO (Floating Production Storage and Offloading) and PS (Platform Supply) vessels, subjected to Jonswap spectrum waves [44] acting mainly on the roll and pitch axis of the vessel per sim-

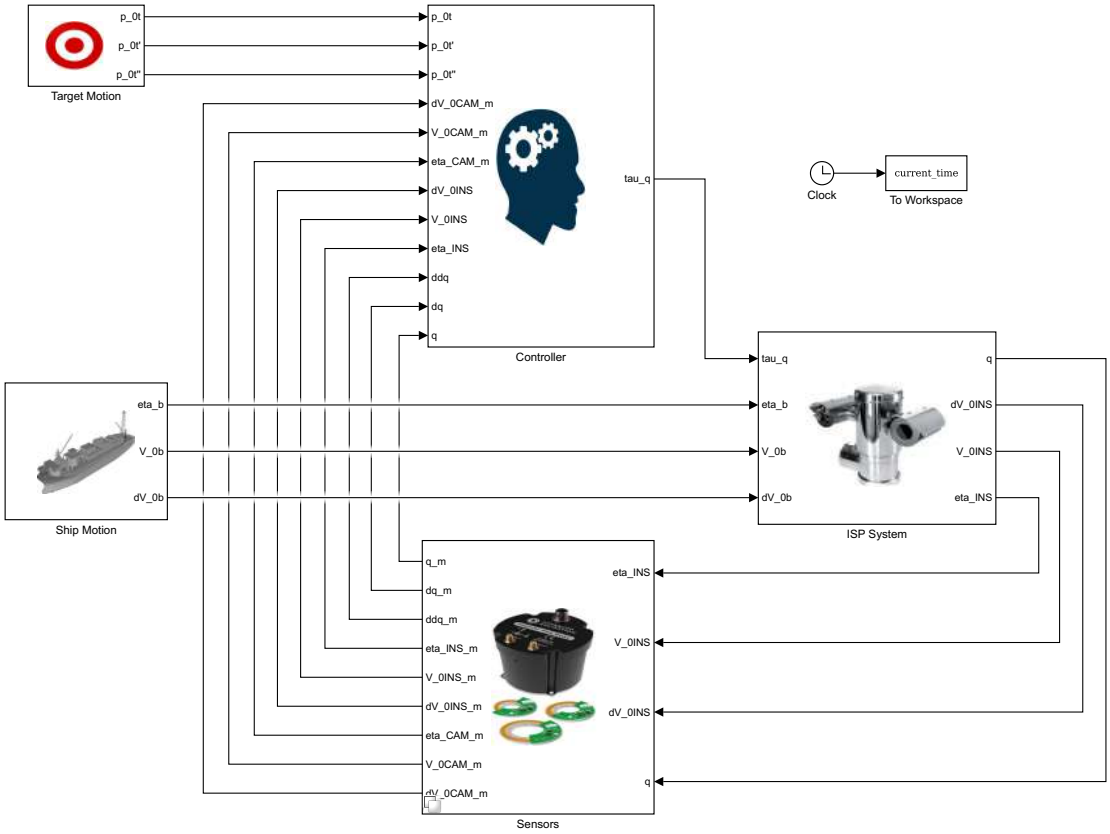


Figure 3.5: Matlab simulator, used for testing all proposed controllers and configurations.

ulation. They were provided by the Tecgraf Institute of Technical and Scientific Software Development from PUC-Rio university, in Brazil. Here, we used the FPSO data with 200 harmonics, 3m height, 10s time period, as shown in Fig. 3.6.

The INS measures either η_0, V_0^0, $\dot{V}_0^0$ for the indirect stabilization case or η_c, V_c^c, $\dot{V}_c^c$ for the direct case. By means of Algorithm 1 and 2, it is possible to use η_0, V_0^0, $\dot{V}_0^0$ to compute the variables of interest η_c, V_c^c, $\dot{V}_c^c$ (indirect configuration). However, computing the motion of $\mathbf{E}_c$ through measurements in $\mathbf{E}_0$ introduces errors in the propagation due to kinematic parameter *uncertainty* and sensor *noise*, resulting in a measurement offset or bias. In the direct case, these propagation errors are absent, at the cost of having a higher payload and size of the ISP.

Remark 16. *In practice, in the direct configuration, the INS and the camera frames will not be perfectly aligned, which can affect the measurements. However, for simplicity, we assume here that the two frames are coincident.*

Table 3.1 contains the kinematic and dynamic parameters used in the simulations. These parameters were obtained from the mechanical design of a 3-DOF ISP currently being developed by a collaboration among several groups, including the

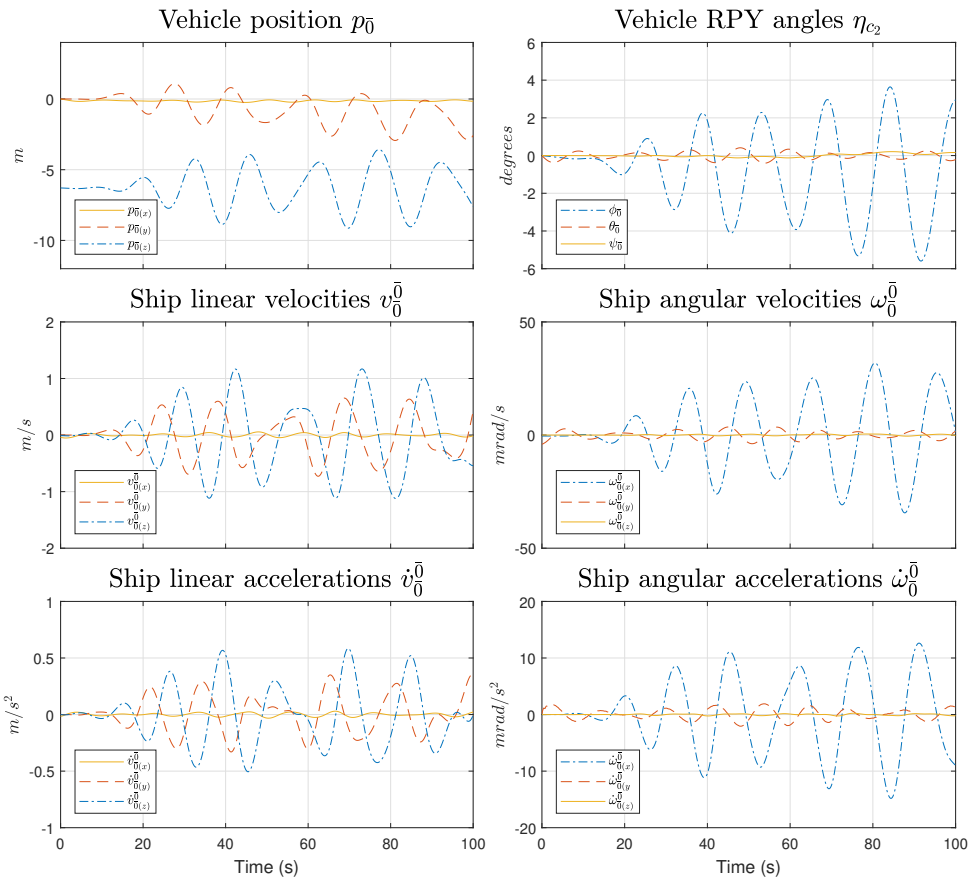


Figure 3.6: Vehicle CG motion data.

Table 3.1: Kinematic and dynamic model parameters, in SI units.

Parameter	$i=1$			$i=2$			$i=3$			Units
	x	y	z	x	y	z	x	y	z	
$p_{i\bar{i}}^{i}$	0.006	0.023	0.326	-0.094	0.006	0.059	0.336	0.006	-0.023	m
$p_{i-1,i}^{i-1}$	0.3	0	0	0	0	0.436	-0.254	0	0	m
$I_{\bar{i}}^{\bar{i}}$	2.42	0.58	1.93	1.12	0.92	0.88	0.54	0.93	0.86	$kg\,m^2$
h_i^{i-1}	$R_{off}\,z_0$			$R_{off}\,y_0$			$R_{off}\,x_0$			m
m_i	18.9			21			26.5			kg

authors laboratory (LEAD/COPPE/UFRJ). Figure 3.7 illustrates a CAD model of the ISP in a graphical simulator also developed by the Tecgraf Institute in an Unity/Gazebo environment.

The alignment error considered in the axis vectors h_i^{i-1} is given by the offset RPY rotation matrix R_{off} with an angle α of assembly misalignment in roll, pitch and yaw axes for each joint. The inertia tensor represented in $\mathbf{E}_i$ can be computed by the *Huygens-Steiner theorem* in (2.1).

The controllers presented in this chapter were tuned using the pole placement

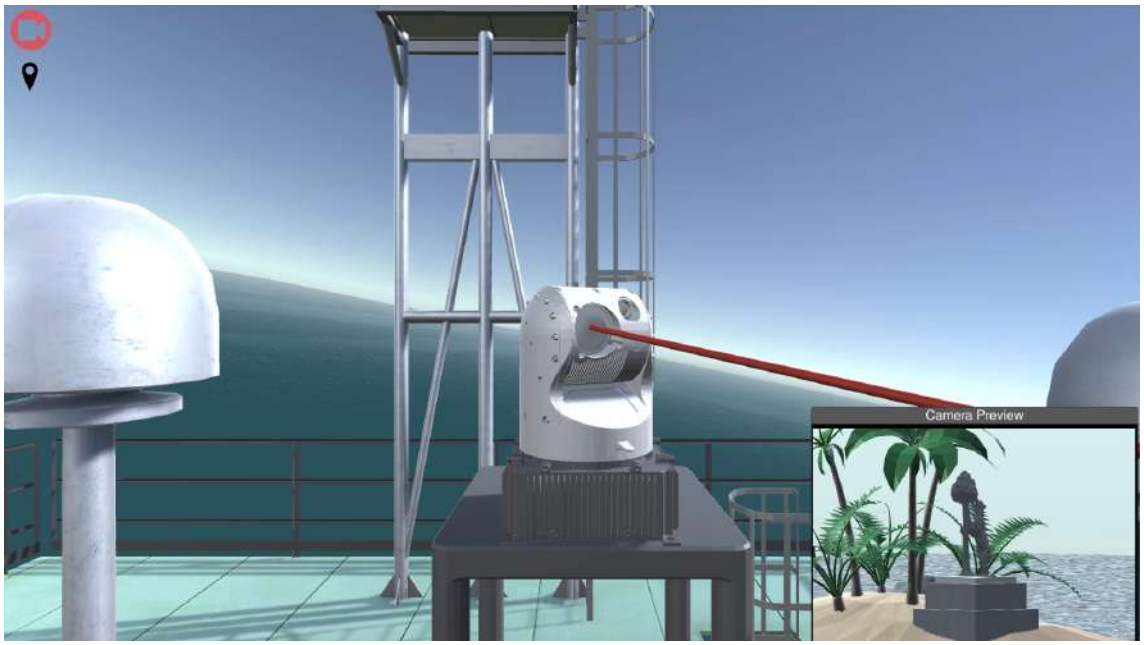


Figure 3.7: Simulator developed by Tecgraf in Unity/Gazebo environment.

method. The desired third-order, closed-loop characteristic equation was chosen to have a distant pole in $-10\,rad/s$, damping ratio $\xi_n = 1.1$ and natural frequency $\omega_n = 2\,rad/s$ for the remaining poles. For the CT-PID methods, $\widehat{M}_\eta$ and $\widehat{\tau}_d^c$, as long as $\widehat{M}_{qq}^0$, $\widehat{\tau}_d^0$, $\widehat{J}_q$, $\widehat{J}_\omega$, $\widehat{L}_q$ and $\widehat{L}_\omega$ were computed using MATLAB® mex files built using the NE algorithm described in Section 2.3.2. The values for the nominal parameters were set as the real values in Table 3.1 and α in R_{off} with a percentage of parametric error, from 0% to 70%.

Both fixed and mobile target configurations will be simulated. For a fixed target, $p_t(t)$, $\dot{p}_t(t)$ and $\ddot{p}_t(t)$ are given by the states of a critically damped second-order filter, whose initial position state $p_t(0)$ is a point far away in the direction of the camera optical axis. This guarantees a smooth RPY reference for the controllers. For a mobile target, its position reference $p_t(t)$ is given by in the circular pattern

$$p_t(t) = p_{t0} + \frac{R\,t}{t+\epsilon} \begin{bmatrix} cos(wt) \\ sin(wt) \\ 0 \end{bmatrix}. \tag{3.22}$$

where $\epsilon > 0$ is a small non-zero parameter. Note that $\lim_{t\to 0} p_t(t) = p_t(0) = p_{t0}$ and $\lim_{t\to\infty} p_t(t)$ is the expression of a planar circle with radius R and centered in $p_{t0} \in \mathbb{R}^3$. This particular function produces a smooth RPY reference, as long as p_{t0} is chosen to be equal to a point far away in the direction of the camera optical axis. The target velocities and accelerations are simply the first and second time derivatives of (3.22), respectively. Figure 3.8 shows a video simulator developed

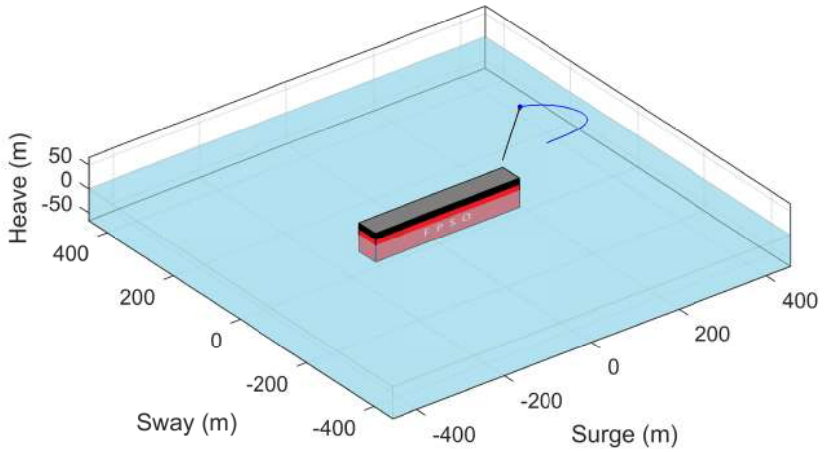


Figure 3.8: Video simulator developed in Matlab, for visualization of the tracking performance.

in Matlab for visualization of the tracking performance, where the ISP optical is following the target described by (3.22).

3.3.1 P-PI Results

Here, the results for the P-PI control (3.6),(3.7) of Section 3.1 are shown. As explained, it was tuned according to the closed-loop characteristic equation (3.8). Figure 3.9 shows the transitory and steady state results for the regulation problem, were the target is fixed at $p_t = \begin{bmatrix} 100 & 100 & 0 \end{bmatrix}^{\mathsf{T}}$. The RPY error showed in Fig. 3.9 is an *ideal* error, computed with respect to the estimated GPS positioning error, and therefore carries the GPS uncertainty in it. This means that, in practice, there will be a steady state offset in the *real* RPY error due to the GPS uncertainty, since the camera position is used to compute the orientation reference on Section 2.2.4. However, since this error cannot be compensated by control, it was omitted here.

The pitch and yaw transient RPY errors in Fig. 3.9 are significantly high, and the norm of the steady state RPY error (jitter) is limited to less than 0.3 degrees. The norm of the joint torques is limited to $5\,Nm$.

Figure 3.10 shows the transitory and steady state results for the tracking problem, where the target motion reference is given by (3.22), with $R = 25\,m$, $w = 2\,\pi/100$, $\epsilon = 1$ and $T = 100\,s$. The center of the circle p_{t0} is chosen as a point $100\,m$ away from the camera position, on the direction of its optical axis. The joint

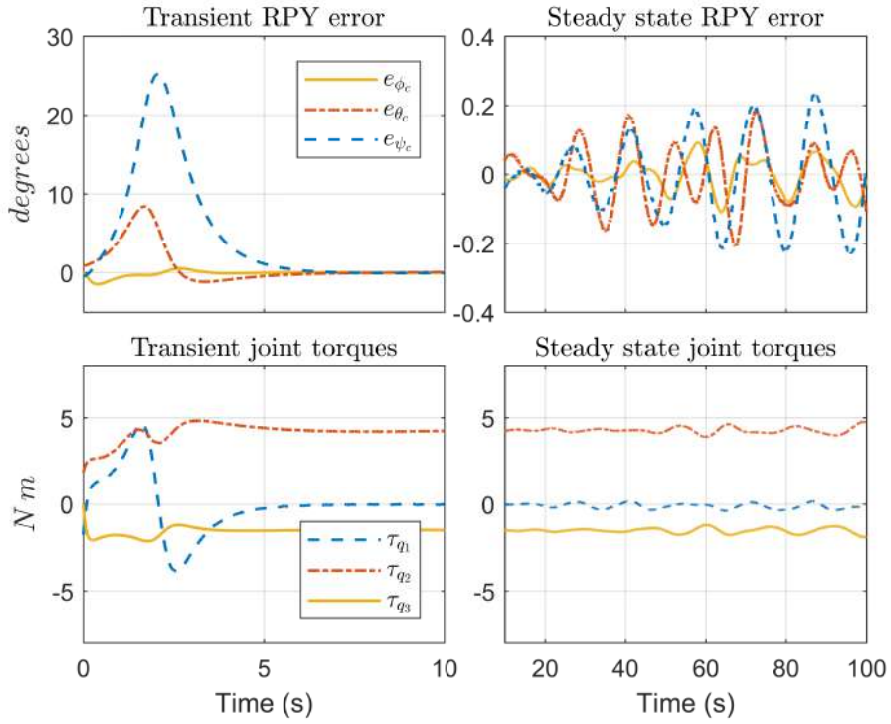


Figure 3.9: Simulation results for the P-PI controller with a fixed target.

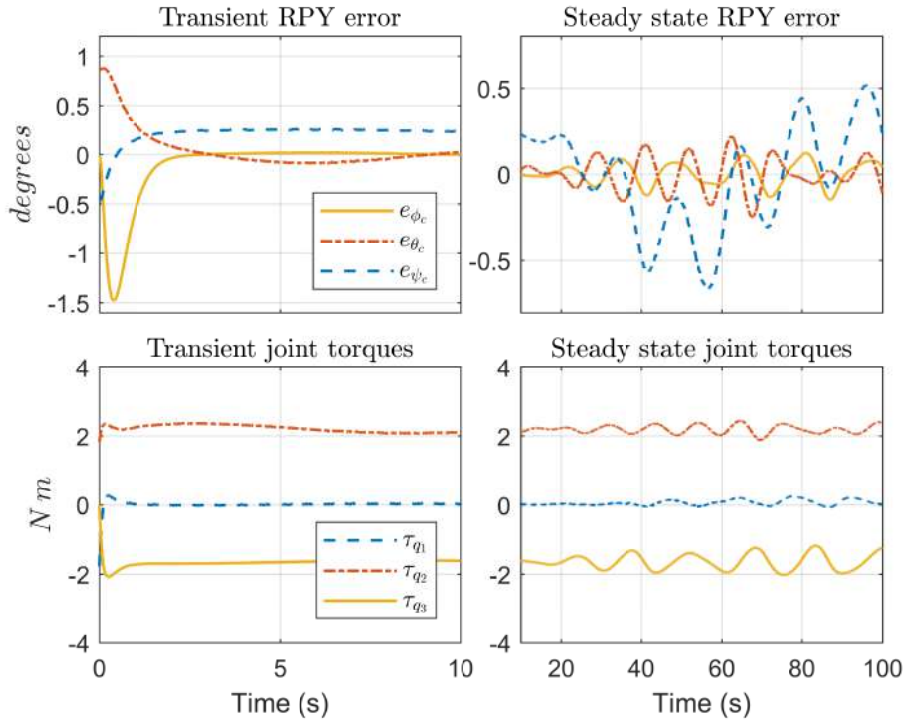


Figure 3.10: Simulation results for the P-PI controller with a mobile target.

torques are still limited under the actuator saturation limits, but the RPY jitter is larger. Simulating more extreme ship conditions (such as higher amplitudes and

frequencies on the sea waves) results in a further increase in the RPY jitter.

3.3.2 Direct Computed-Torque Results

In this section, we present the simulation results for the CT-PID control law (3.14), (3.15) proposed in Section 3.2.1, considering the *direct configuration* for the inertial sensors. In this case, the computed-torque controller depends on the full set of ISP dynamic and geometric parameters. Therefore, some simulations are made considering full and uncertain knowledge of these parameters, for both fixed and mobile targets. The controller PID gains were tuned by the pole placement method described in Section 3.2.1.

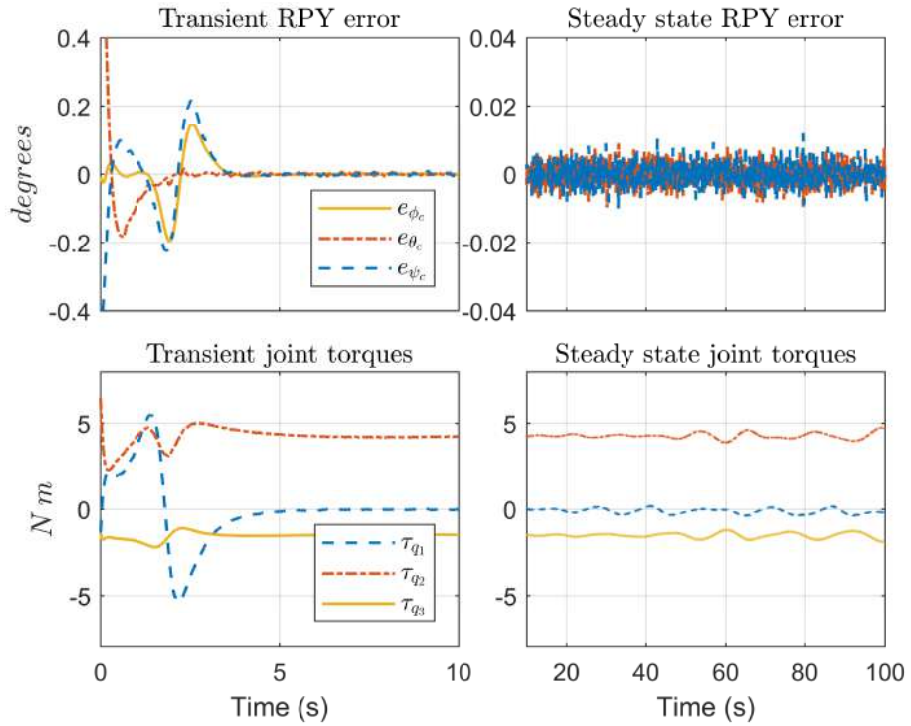


Figure 3.11: Simulation results for the direct CT-PID controller with a fixed target and full knowledge on the ISP parameters.

The simulation results for a fixed target at $p_t = \begin{bmatrix}100 & 100 & 0\end{bmatrix}^\mathsf{T}$ is shown in Fig. 3.11. Considering full knowledge of the system parameters, the RPY error tends to a very small residue of 0.02 degrees, due to sensor hardware disturbances and joint velocity/acceleration estimation noise. The behavior of the joint torques is similar to the P-PI controller. Similar results are obtained for a mobile target, in Fig. 3.12. Unlike the P-PI controller, there is no performance degradation for tracking control is this case.

We expect the RPY jitter to increase when some level of uncertainty is present

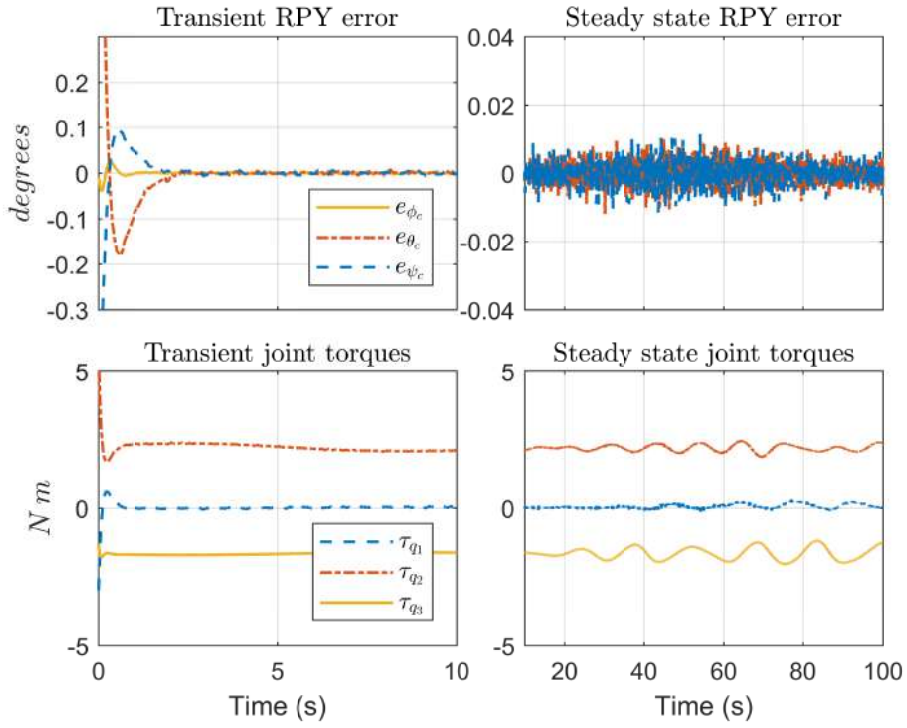


Figure 3.12: Simulation results for the direct CT-PID controller with a mobile target and full knowledge on the ISP parameters.

on the ISP dynamic parameters. Therefore, the next step is adding some level of uncertainty on the ISP parameters used in the controller. Figure 3.13 shows the results obtained for 20% of error under the ISP dynamic parameters Π_d, and full knowledge of the ISP geometric parameters Π_g, according to Assumption (i) from Theorem 4. Notice that, in terms of jitter, the proposed controller still overcomes the performance of the P-PI controller, while maintaining an acceptable joint torque response. For the mobile target simulation, the RPY error is very similar to Fig. 3.13, while the joint torque response is similar to Fig. 3.12. Again, unlike the P-PI controller, no performance degradation was observed in the case of tracking control.

For higher levels of uncertainty on Π_d, jitter performance keeps degrading until it reaches the P-PI control performance. For even higher levels of uncertainty, P-PI control finally outcomes the CT-PID.

3.3.3 Indirect Computed-Torque Results

In this section, we present the simulation results for the CT-PID control proposed in Section 3.2.2, considering the *indirect configuration* for the inertial sensors. Here, the control law is given by (3.19) and (3.21). The immediate advantage of this approach is the lack of dependency on joint acceleration estimations; therefore, the jitter level

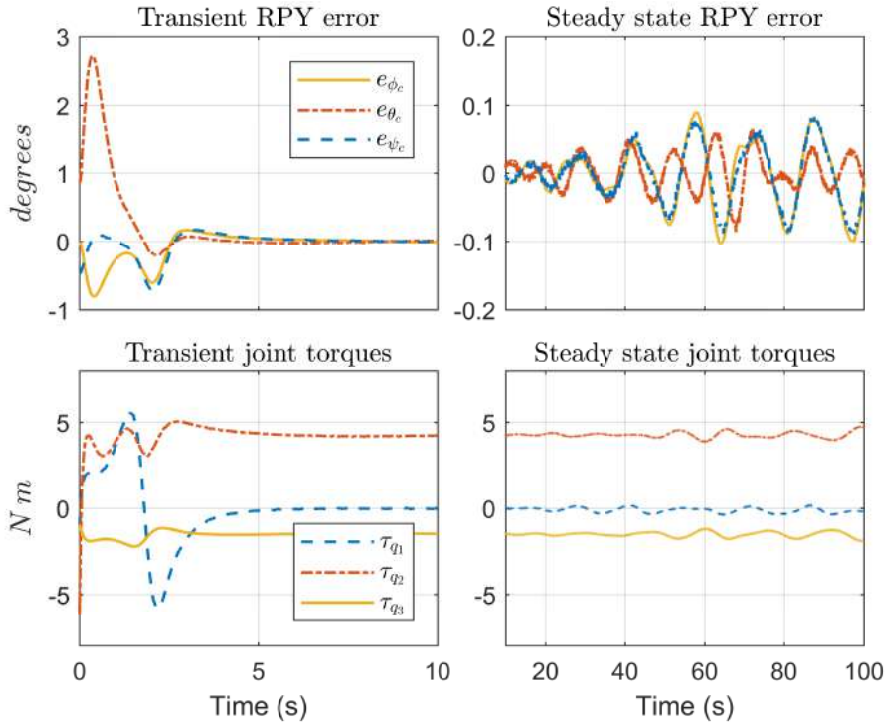


Figure 3.13: Simulation results for the direct CT-PID controller with a fixed target under 20% of error on Π_d (Π_g is known).

is expected to decrease. The next advantage is the possibility of investigating the effects of uncertainty on the geometric parameters as well. Once again, the controller PID gains were tuned according to the method described in Section 3.2.2.

Figure 3.14 shows the results for regulation control and full knowledge of the ISP parameters. The transient response is slightly worse, with higher RPY error overshoot and high frequency oscillations on the joint torques. However, the RPY jitter level and the steady state joint torques seem to be similar to the direct case in Fig. 3.11. The results for the mobile case are very similar to Fig. 3.12.

Next, we once again add a level of uncertainty on the ISP parameters used in the controller. Notice that in this case, parametric errors on the *dynamic* parameters Π_d affect only the inner controller (3.19), while parametric errors on the *geometric* parameters Π_d affect the outer controller (3.21) only. As before, starting with a parametric error of 20% on the *dynamic* parameters Π_d yields Fig. 3.15. The steady state result is very similar to Fig. 3.13, and as before, the transient response is more oscillatory. The response for 20% of error on the *geometric* parameters Π_g is shown in Fig. 3.16. As observed, the RPY jitter response is slightly smaller than in Fig. 3.15. However, since an error on Π_g introduces propagation errors for the computation of the camera orientation in the indirect configuration, a small *offset* is expected to be present in the RPY error. This offset is more evident under higher

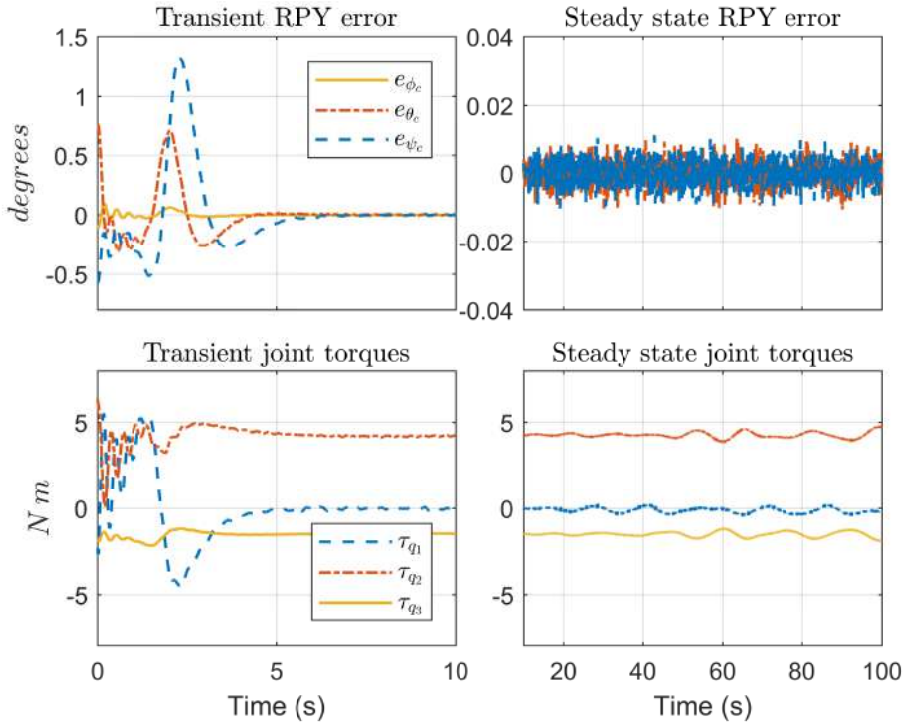


Figure 3.14: Simulation results for the indirect CT-PID controller with a fixed target and full knowledge on the ISP parameters.

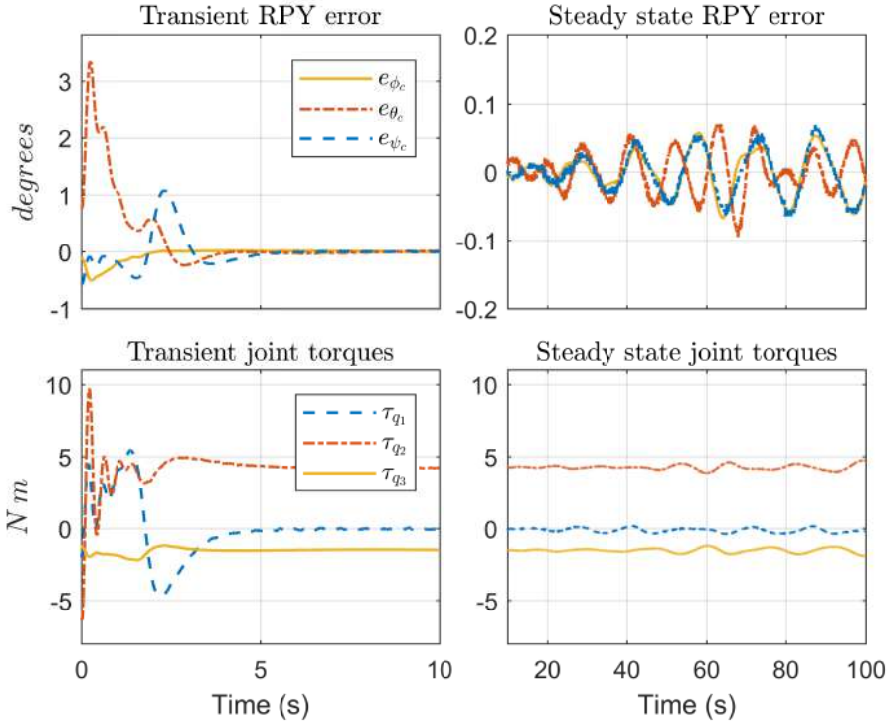


Figure 3.15: Simulation results for the indirect CT-PID controller with a fixed target under 20% of error on Π_d (Π_g is known).

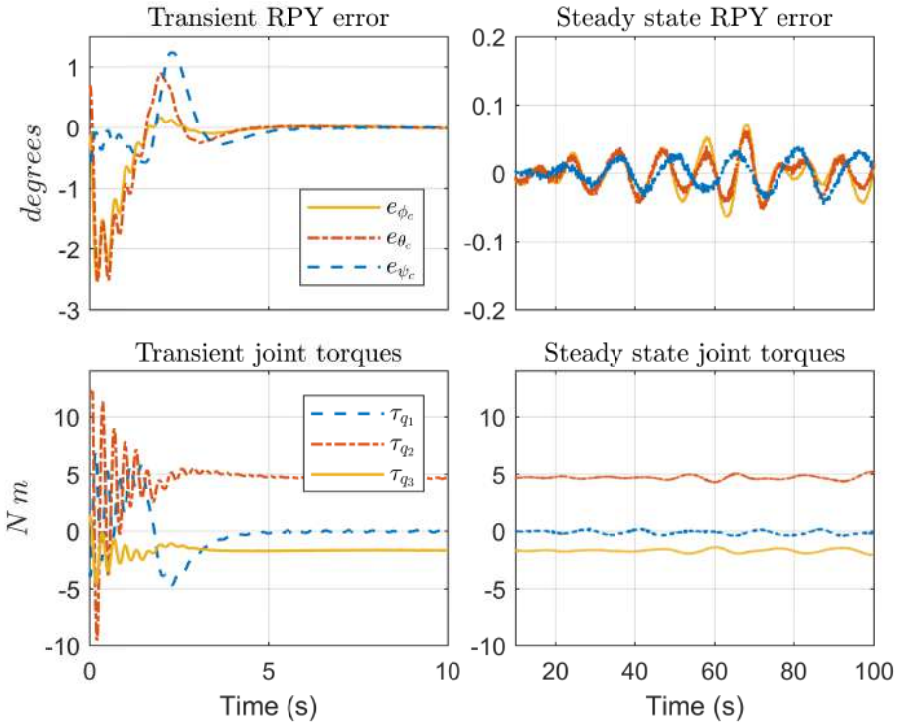


Figure 3.16: Simulation results for the indirect CT-PID controller with a fixed target under 20% of error on Π_g (Π_d is known).

values of geometric uncertainty, as illustrated in Fig. 3.17, which shows the results for 50% of error on Π_g.

For higher uncertainties on both Π_d and Π_g, the RPY jitter keeps increasing, as expected by Theorem 5. The offset on the RPY error however, only increases as the parametric error on Π_g increases, which is an expected result.

3.3.4 Robustness Analysis

As exposed, the performance of the CT-PID controllers is conditioned to a satisfactory level of knowledge of the ISP dynamic and geometric parameters Π_d and Π_g. Although *identification* techniques could be employed to obtain such level of parameter certainty, these techniques can be affected by numerous sources of errors, such as sensor noise, bias and quantization. Therefore, it is useful to perform a practical robustness analysis of the proposed CT-PID controllers with respect to uncertainties on the ISP parameters. This analysis is useful not only to set appropriate uncertainty bounds on the identification techniques that must be employed to reach a certain performance, but also to obtain insight about the differences among the direct and indirect configurations, and about how the CT-PID control performance can be properly compared with the P-PI controller.

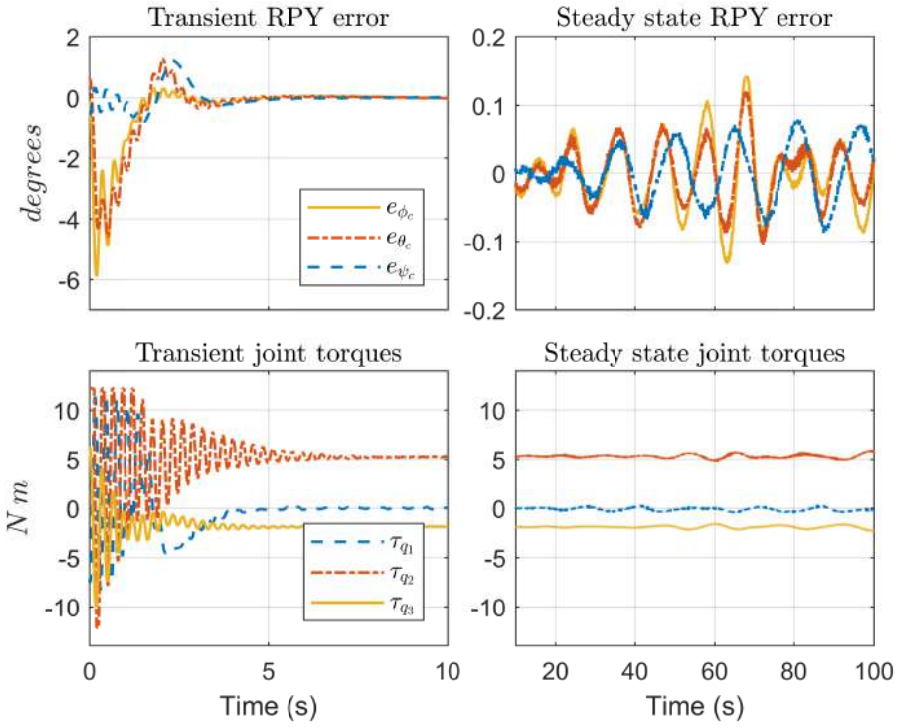


Figure 3.17: Simulation results for the indirect CT-PID controller with a fixed target under 50% of error on Π_g (Π_d is known).

Therefore, this section presents a practical robustness analysis on the RPY error with respect to the ISP parameters. It attempts to confirm the main conclusions about the *jitter/offset* duality so far, and how they are related to Π_d and Π_g.

Three sets of simulations were performed for each inertial sensor configuration (direct or indirect):

1. 8× simulations with error on Π_d varying from 0 to 70% and fully known Π_g;
2. 8× simulations with error on Π_g varying from 0 to 70% and fully known Π_d;
3. 8× simulations with error on both Π_d and Π_g varying from 0 to 70%.

The RPY error considered on these simulations was the *real* RPY error, computed using the real camera position, without GPS uncertainty. This way, the effects of the offset bias can be properly observed. The results for the RPY error jitter and offset (bias) obtained for each one of the total of 48 simulations are summarized and discussed in the next graphs.

The real jitter and bias on the RPY errors for the first set of simulations are illustrated in Fig. 3.18. Notice how the jitter level is bounded by the norm of $\widetilde{\Pi}_d$, in perfect accordance to Theorems 4 and 5. Ideally, the jitter levels at 0% of parametric error should be zero, but in practice they are affected by sensor hardware

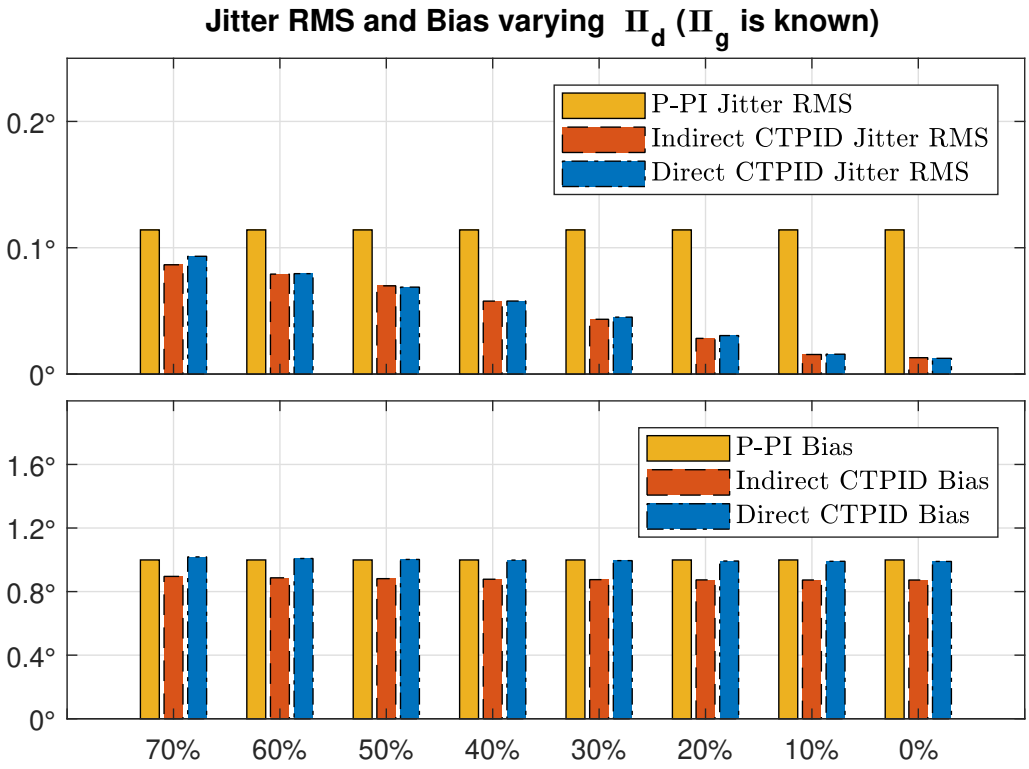


Figure 3.18: Jitter and offset on the RPY error for variations on Π_d (Π_g is fixed).

disturbances, as observed. For all simulations, the proposed CT-PID controller on both direct and indirect configurations outcomes the P-PI performance in terms of RPY jitter. Actually, the performance is better than the P-PI for up to 80% of dynamic parametric error. It is worth noticing that, in terms of jitter, there is no significant difference between the direct and the indirect configurations when the geometric parameters are fully known.

With respect to RPY offset, there are some interesting conclusions. By Fig. 3.18, it seems that the dynamic parameters do not have any influence on the RPY offset. This is an expected conclusion considering the *real* RPY errors, since the LOS reference computation suffers from GPS uncertainty. In the indirect case, the LOS reference computation is also dependent on the geometric parameters of the ISP, which are supposed to be fully known.

Next, the results obtained for the second set of simulations are shown in Fig. 3.19. Clearly, geometric parameter errors affect both RPY jitter and offset, for both control configurations. However, their impact in the RPY jitter for the indirect case is much higher than in the direct case. This is expected due to imperfect canceling of the nonlinear kinematic terms by (3.21). Even so, the CT-PID controllers have better jitter performance than the P-PI for up to 30% of error in Π_g.

Notice how, although the computed-torque expression of the direct CT-PID controller (3.15) is dependent on both dynamic and geometric parameters, the sensi-

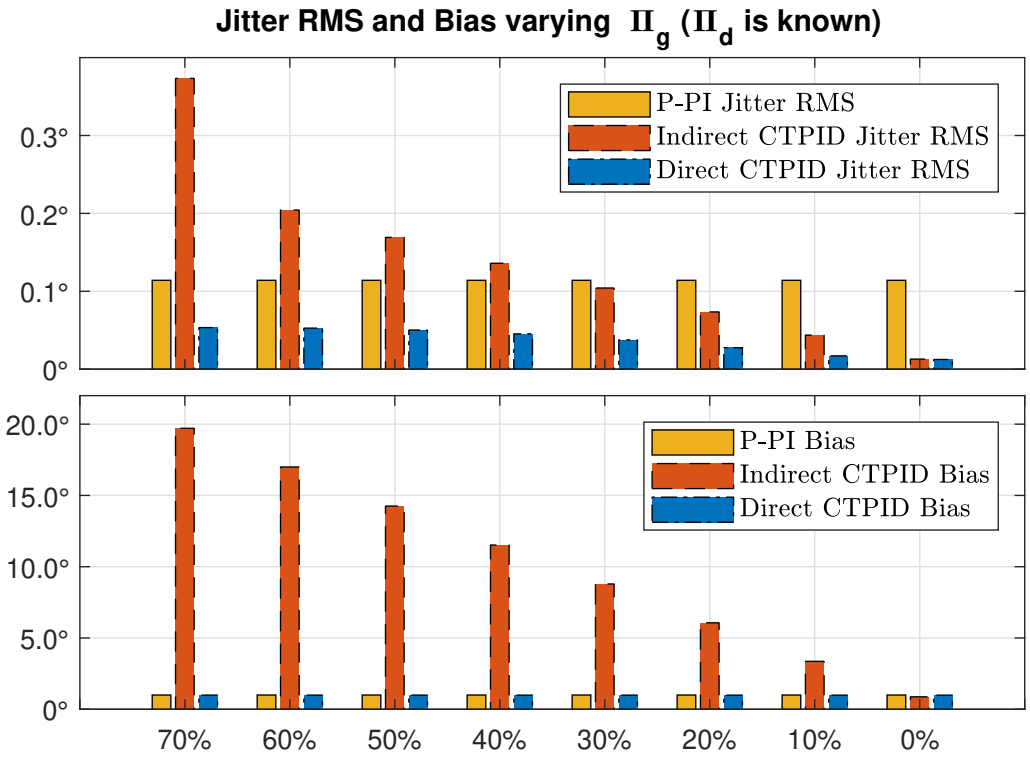


Figure 3.19: Jitter and offset on the RPY error for variations on Π_g (Π_d is fixed).

bility of the RPY jitter to geometric parameter errors is much smaller in the direct approach. Actually, it is even better than the P-PI for all simulations.

This time, the offset error on the indirect CT-PID also increases for larger values of geometric uncertainty. This is mostly due to propagation of errors in the LOS reference computation, since the integral term on the PID controller is able to remove the offset due to imperfect canceling of the nonlinear terms.

Lastly, the results obtained for the third set of simulations are shown in Fig. 3.20. Here, the performance of the controllers is studied considering errors in all parameters. Once again, the RPY jitter for the CT-PID controllers is smaller than the P-PI for up to 30% of parametric error. The RPY jitter for the CT-PID controllers in the third set of simulations was actually slightly smaller than for the other simulations, an unexpected result that does not seem trivial to explain. In the case of RPY offset, the errors due to the kinematic parameters dominate, and the results are virtually indistinguishable from Fig. 3.19.

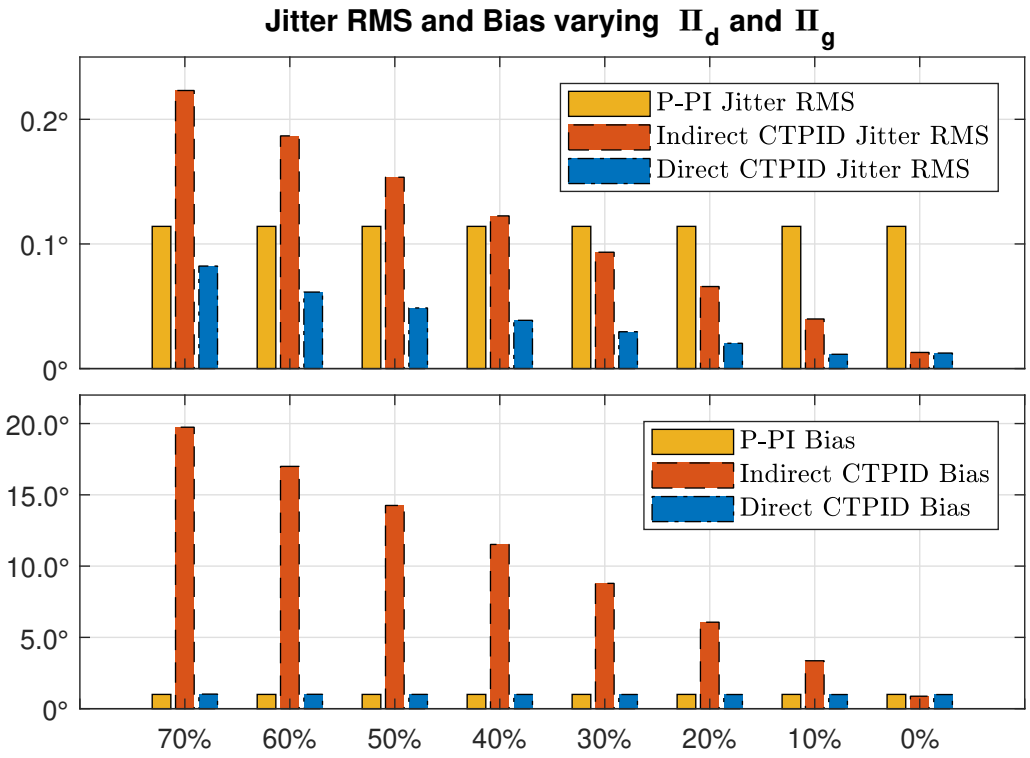


Figure 3.20: Jitter and offset on the RPY error for variations on both Π_d and Π_g.

Chapter 4

Super-Twisting Control

In this chapter, we introduce two controllers based on the Super Twisting Algorithm. They attempt to improve even further the jitter/bias performance obtained by the Computed Torque scheme presented in the previous chapter, focusing on achieving the desirable characteristics of Sliding Mode Control: finite-time convergence for the sliding surfaces and ideal rejection of matched disturbances.

Furthermore, we also formulate the orientation error in *unit quaternions*, attempting to solve the problem of representation singularities that arises in all controllers based in minimal representations for orientation, such as the RPY angles.

First, a brief survey on Sliding Mode Control is presented on Section 4.1, focusing on its main characteristics, applications and limitations. Then, the proposed controllers are presented and their stability properties are studied in Section 4.2, considering two distinct cases: (i) full state feedback and (ii) output feedback with a Higher Order Sliding Mode (HOSM) observer. Finally, simulation results are illustrated and discussed in Section 4.3.

4.1 Sliding Mode Control

The main idea of sliding mode control (SMC) is to chose an appropriate *constraint* for the system states that encapsulates a *desired behavior* on its *dynamics*, and design a control law that ensures that this constraint is satisfied by using the phenomena known as sliding mode, or dynamical collapse [9]. Consider a general non-linear state-space system dynamics

$$\dot{x} = f(x, u, d)\,, \tag{4.1}$$

where $x \in \mathbb{R}^n$ is the state vector, $u \in \mathbb{R}^m$ is a control input and $d \in \mathbb{R}^q$ represents an external bounded disturbance vector. Also, assume that $f(\cdot)$ is differentiable with respect to the states x and absolutely continuous with respect to time. Then, define a surface in the state-space given by

$$\mathbb{S} = \{x \,|\, \sigma(x) = 0\}\,. \tag{4.2}$$

Definition 1 (Ideal sliding mode)**.** *An ideal, first-order sliding mode (FOSM) is taking place on* (4.1) *if the states* $x(t)$ *evolve in time such that* $\sigma(x(t)) = 0 \;\forall\; t > t_r$*, for a finite* $t_r \in \mathbb{R}^+$ *such that* $\sigma(x(t_r)) = 0$*.*

During sliding mode, the system state order is reduced, and the system trajectories are all confined to the set $\mathbb{S}$. If the the control input is given in the form of a *discontinuous* control law $u(x)$, the closed-loop system dynamics is given by

$$\dot{x} = f^c(x, d)\,, \tag{4.3}$$

where $f^c(.)$ is *discontinuous* with respect to x. In the classical theory of differential equations, Lipschitz assumptions under function $f(.)$ are employed to guarantee *uniqueness* of the solutions [8]. Therefore, the solutions of a differential equation with a discontinuous right-hand side are from now on understand in the sense of Filippov, meaning that they are constructed as an average of the solutions obtained from approaching the point of discontinuity from different directions.

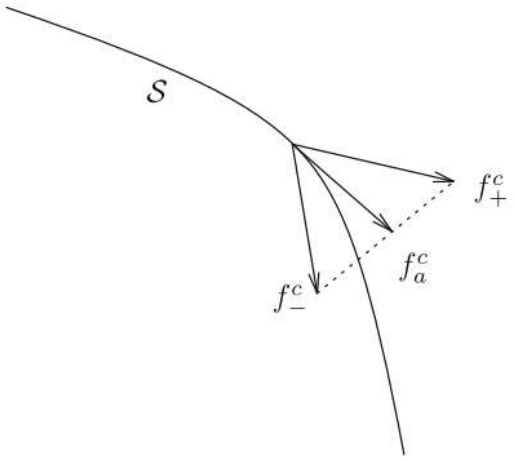


Figure 4.1: A schematic of the Filippov solution, from [9].

Suppose x_0 is a point of discontinuity on $\mathbb{S}$ and define f_-^c, f_+^c as the limits of $f^c(x)$ as the point x_0 is approached from opposite sides of the tangent to $\mathbb{S}$ at x_0.

The solution proposed by Filippov is

$$\dot{x} = (1-\alpha) f_{-}^{c}(x) + \alpha f_{+}^{c}(x)\,, \qquad 0 < \alpha < 1\,, \tag{4.4}$$

where the scalar α is chosen so that $f_a^c = (1-\alpha) f_{-}^{c}(x) + \alpha f_{+}^{c}(x)$ is *tangential* to $\mathbb{S}$, as illustrated in Fig. 4.1. Therefore, defining the convex set

$$F(x) = \{\ (1-\alpha) f_{-}^{c}(x) + \alpha f_{+}^{c}(x)\ \forall\ \alpha \in [0\,,1]\ \}\,, \tag{4.5}$$

equation (4.4) could be though of as a *differential inclusion* instead of a differential equation itself, represented by

$$\dot{x} \in F(x)\,. \tag{4.6}$$

Suppose that the trajectories of system (4.1) are following an *ideal* sliding mode. Therefore, since $\sigma(x) = 0$, the time derivative of the sliding variable σ must be *ideally* zero to maintain the trajectories of the system at $\sigma(x) = 0$. Then, there exists a theoretical control $u_{eq}(t)$ such that

$$\dot{\sigma} = \frac{\partial \sigma}{\partial x}\,\dot{x} = \frac{\partial \sigma}{\partial x}\, f(x, u_{eq}, d) = 0\,, \tag{4.7}$$

which is known by the *equivalent* control, by Utkin [45]. For example, consider that system (4.1) can be written in the affine form

$$\dot{x} = f(x) + g(x)\, u + d\,. \tag{4.8}$$

In this case, (4.7) can be rewritten as

$$\frac{\partial \sigma}{\partial x}\, f(x) + \frac{\partial \sigma}{\partial x}\, g(x)\, u_{eq} + \frac{\partial \sigma}{\partial x}\, d = 0\,, \tag{4.9}$$

which in turn, makes us able to compute an expression for the equivalent control:

$$u_{eq} = -\left(\frac{\partial \sigma}{\partial x}\, g(x)\right)^{-1} \frac{\partial \sigma}{\partial x}\, f(x) - \left(\frac{\partial \sigma}{\partial x}\, g(x)\right)^{-1} \frac{\partial \sigma}{\partial x}\, d\,. \tag{4.10}$$

Notice that (4.10) depends on the disturbance, which is, in general, unknown. Therefore, the equivalent control must be understood as the theoretical control action that will maintain the system in the sliding condition, rather than representing an actual physically implementable control law.

Substituting (4.10) into the affine system (4.8), we get the closed-loop system

$$\dot{x} = \left(\mathbf{I} - g(x) \left(\frac{\partial \sigma}{\partial x} g(x) \right)^{-1} \frac{\partial \sigma}{\partial x} \right) f(x) + \left(\mathbf{I} - \left(\frac{\partial \sigma}{\partial x} g(x) \right)^{-1} \frac{\partial \sigma}{\partial x} \right) d \,. \tag{4.11}$$

Now, suppose that the disturbance $d(t)$ is acting on the same channel as the control input, meaning that

$$d(t) = g(x)\, \epsilon(t) \,, \tag{4.12}$$

and $\epsilon(t)$ is an unknown signal. Then, $d(t)$ is known as a *matched* disturbance. Using (4.12), it is evident that the closed loop system (4.11) becomes

$$\dot{x} = \left(\mathbf{I} - g(x) \left(\frac{\partial \sigma}{\partial x} g(x) \right)^{-1} \frac{\partial \sigma}{\partial x} \right) f(x) \,, \tag{4.13}$$

which is completely independent on the disturbance. This invariance property has motivated research in sliding mode control [9]. From (4.11), the choice of the sliding surface $\sigma(x)$ clearly affects the resulting dynamics of the closed loop system. In terms of control system design, this selection is one of the key design choices.

Sliding modes are naturally classified by the smoothness degree of the constraint function calculated along the system trajectories. Let the constraint be of the conventional form $\sigma(x,t) = 0$, where $\sigma(\cdot)$ is some smooth function of the state and time, and suppose a discontinuous dynamic system is following Filippov trajectories under this constraint (i.e., a first order sliding mode is occurring). The sliding mode $\sigma = 0$ may be classified by the first total derivative $\sigma^{(r)}$ which contains a discontinuity in a small vicinity of the sliding mode trajectories. Number r is called the *sliding order*.

Definition 2 (r-sliding mode)**.** *Consider a discontinuous, Filippov differential inclusion* (4.6) *with a smooth output function* $\sigma(x)$ *and let it be understood in the Filippov sense. Then, if*

1. *the total time derivatives* $\sigma, \dot{\sigma}, \ldots, \sigma^{r-1}$ *are continuous functions of* x*,*
2. *the set*

$$\sigma = \dot{\sigma} = \ldots = \sigma^{r-1} = 0 \,, \tag{4.14}$$

   is a nonempty integral set (i.e., consists of Filippov trajectories),

3. *the Filippov set of admissible velocities at the* r*-sliding points* (4.14) *contains more than one vector,*

then, the motion on the r*-sliding set* (4.14) *is said to exist in an* r*-sliding mode [9].*

For $r = 1$, then Definitions 2 and 1 are equivalent. For $r > 1$, the correponding r-sliding mode is also known as a higher order sliding mode (HOSM).

4.2 Super-Twisting Controller with Quaternion Feedback

In this section, a second-order sliding mode (SOSM) controller based on the *super-twisting algorithm* (STA) will be developed for the stabilization and tracking of the ISP. According to Definition 2, in this context *second-order* means that the sliding variable itself and its first derivative are continuous, and the controllers are able to achieve *second-order sliding motion* (SOSM), meaning that both the sliding variable itself and its first derivative are ideally zero during sliding. Two cases are considered: super-twisting control (STC) with full state feedback and with output feedback only.

Defining the joint angle and velocity states $x_1 = q \in \mathbb{R}^3$ and $x_1 = \dot{q} \in \mathbb{R}^3$, the dynamic model (3.17) can be rewritten as:

$$\begin{aligned} \dot{x}_1 &= x_2 \,, \\ \dot{x}_2 &= {M_{qq}^0}^{-1} \tau_q + x_3(x_1, x_2, \Pi, t) \,, \end{aligned} \tag{4.15}$$

where $x_3 = -(M_{qq}^0)^\dagger \, \tau_d^0$ is a state-dependent disturbance.

Remark 17. *Note that, under assumption of torque control $u(t) = \tau_q$, state-space model* (4.15) *is a double-integrator with a nonlinear high-frequency gain and a matched disturbance τ_d^0.*

Now, it is possible to rewrite (2.54):

$$\dot{\omega}_c^c = J_{0c_2}^c(q, \Pi_g) \, \ddot{q} + \dot{J}_{0c_2}^c(q, \dot{q}, \Pi_g) \, \dot{q} + \dot{\omega}_0^c$$

in a state space form using the *quaternion propagation* formula in (A.18), yielding:

$$\begin{aligned} \dot{y}_1 &= \frac{1}{2} \, \mathbf{h}_+(y_1) \, y_2 \,, \\ \dot{y}_2 &= J_{0c_2}^c(x_1, \Pi_g) \, \dot{x}_2 + y_3(x_1, x_2, \Pi_g, t) \,. \end{aligned} \tag{4.16}$$

where the state $y_1 = \overline{r}_c = \begin{bmatrix} y_{11} & y_{12}^\mathsf{T} \end{bmatrix}^\mathsf{T}$ is the vector representation of the camera orientation $r_c \in \mathbb{H}^*$, with $y_{11} = \eta_c$ and $y_{12} = \epsilon_c$ being the scalar and vector components, respectively. State $y_2 = \omega_c^c \in \mathbb{R}^3$ is the camera body angular velocity, while $y_3 = \dot{J}_{0c_2}^c(x_1, x_2) \, x_2 + \dot{\omega}_0^c(x_1, x_2, t) \in \mathbb{R}^3$ is another state-dependent disturbance, given by (2.54).

Remark 18. *Note that* (4.16) *is a double integrator with a nonlinear high-frequency gain and a matched disturbance* y_3 *with respect to a control input* $\dot{x}_2$.

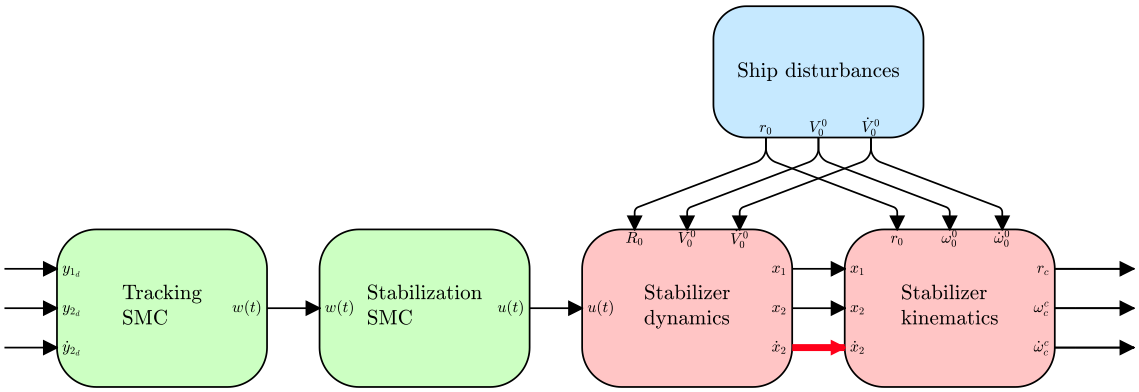


Figure 4.2: Block diagram for the open-loop system with a cascade controller.

This structure strongly suggests the use of a *cascade controller* for both stabilization and tracking. An inner controller acts on the torque input $u(t)$ in (4.15) to control the angular acceleration $\dot{x}_2$, providing dynamic stabilization for the system, while an outer tracking controller acts on the joint angular acceleration reference for $\dot{x}_2$, controlling the camera orientation y_1. Figure 4.2 illustrates the proposed control scheme in a block diagram form.

Notice that the *equivalent* control associated to the stabilization system is

$$u_{eq}(t) = M^0_{qq}(x_1)\, w(t) + \tau^0_d \,. \tag{4.17}$$

In other words, this is the control signal that perfectly rejects the disturbances in (4.15), and performs the ideal tracking of $\dot{x}_2$ into a general *joint acceleration reference* $w(t)$. This is the stabilization control law proposed in (3.19), but with perfect knowledge of the ISP parameters Π_g and Π_d. However, in this case, there are no sensors or exact dynamic model to measure the state and time-dependent disturbance τ^0_d.

From the developments of Section 2.2.4, given an orientation reference $r_{c_d}(t) \in \mathbb{H}^*$ and an angular velocity reference for the camera, they can be represented in vector form by

$$\overline{r}_{c_d}(t) = y_{1_d}(t) = \begin{bmatrix} \eta_{c_d}(t) \\ \epsilon_{c_d}(t) \end{bmatrix} = \begin{bmatrix} y_{11_d}(t) \\ y_{12_d}(t) \end{bmatrix} . \tag{4.18}$$

The *desired* angular velocity for the camera is given as $\omega^c_{c_d}(t) = y_{2_d}(t)$, and was also obtained by the methods of Section 2.2.4. Then, the quaternion and angular

velocity errors can be defined as [20]:

$$e_c = r_{c_d}(t) \cdot r_c^* \in \mathbb{H}^* \,, \tag{4.19}$$

$$e_\omega = \omega_{c_d}^c(t) - \omega_c^c \in \mathbb{R}^3 \,. \tag{4.20}$$

Note that when $r_c = r_{c_d}(t)$, the orientation error (4.19) is the *unit quaternion* l_Q.

Remark 19. *This particular form for the quaternion error $e_c \in \mathbb{H}^*$ is widely used in robotics literature, since the error is given on the same space than the orientation variable. In other words, it preserves the group of unit quaternions on the error dynamics. Moreover, it provides better computational precision, and is a more natural way of representing the orientation of a rigid body than minimal representations in $\mathbb{R}^3$ [20].*

Next, we present a theorem about quaternion error stability that will be useful in the following developments.

Theorem 6 (Stability of Quaternion Error Dynamics)**.** *Using the orientation and angular velocity errors as defined in* (4.19), (4.20), *$e_c = \mathsf{l}_Q \in \mathbb{H}^*$ is a globally asymptotically stable equilibrium point of the error dynamic equation*

$$e_\omega + K_c \, \mathsf{Im}(e_c) = 0 \,, \qquad K_c > 0 \,. \tag{4.21}$$

A complete demonstration for Theorem 6 can be found in [46]. A sketch of the proof can also be found in Section B.6.

Inspired by the works of [33, 47] and by the form of the system dynamic equations, two control schemes based on the Super-Twisting Algorithm are presented in the next two subsections. The first one supposes the availability of the full state vector for the dynamic subsystem (4.15) (joint angles and velocities $x_1, x_2 \in \mathbb{R}^3$), while the second employs a Higher Order Sliding Mode (HOSM) observer to estimate the unmeasured joint velocity state $\dot{q} \in \mathbb{R}^3$.

4.2.1 Super-Twisting Controller with Full State Feedback

In this section, the super-twisting algorithm is proposed to both stabilize the ISP and to track a mobile target, under the assumption of full state feedback.

First, define a *stabilizing* sliding variable as

$$s_x = x_2 - \int_0^t w(\tau) \, d\tau \,. \tag{4.22}$$

where $w(t) \in \mathbb{R}^3$ is a *bounded* joint acceleration reference. The objective is to guarantee *SOSM* under s_x for an arbitrary (bounded) signal $w(t)$, such that

$$s_x = \dot{s}_x = \dot{x}_2 - w(t) = 0\,.$$

In other words, the joint accelerations must follow a given bounded reference $w(t)$. This way, the plant dynamics (4.15) is reduced to an ideal double integrator.

Next, let the *tracking* sliding variable be defined as the quaternion error dynamics in (4.21):

$$s_y = e_\omega + K_c\,\mathsf{lm}(e_c)\,, \quad K_c > 0\,. \tag{4.23}$$

The next objective is to guarantee that $s_y = e_\omega + K_c\,\mathsf{lm}(e_c) = 0$ in finite time using continuous control. This will guarantee LOS tracking performance, given that the correct LOS references are given.

The following theorem provides an stability analysis for the proposed sliding mode controller.

Theorem 7 (*Cascade Super-Twisting Controller with Full State Feedback*)**.** *Let* (4.15) *and* (4.16) *be the dynamic and kinematic models of an ISP system under parametric uncertainties. Assume the following:*

(i) the body Jacobian $J^c_{0c_2}(x_1) \in \mathbb{R}^{3\times 3}$ is full-rank.

(ii) the ISP joint velocities $x_2 \in \mathbb{R}^3$ are mensurable and uniformly norm-bounded and joint accelerations $\dot{x}_2 \in \mathbb{R}^3$ are uniformly norm-bounded.

(iii) the zero, first and second-order time-derivatives of the vehicle velocity twist $V_0^0 \in \mathbb{R}^6$ are uniformly norm-bounded.

1. *The tracking sliding surface is defined in* (4.23)*. The corresponding tracking control law is*

$$w(t) = \widehat{J}^c_{0c_2}(x_1)^{-1}\left(\dot{y}_{2_d}(t) + K_c\,\psi + \Lambda_3\lfloor s_y\rceil^{1/2} + \Lambda_4\int_0^t \lfloor s_y\rceil^0 d\tau\right), \tag{4.24}$$

$$\begin{aligned}\psi(y_1, y_2, y_{1_d}) = {} & y_{11}\,\dot{y}_{12_d} - 0.5\,y_{12}^{\mathsf{T}}\,y_2\,y_{12_d} - \dot{y}_{11_d}\,y_{12} - \dot{\widehat{y}}_{12_d}\,y_{12} \\ & - 0.5\,y_{11_d}\,(y_{11}\,\mathsf{I}_3 - \widehat{y}_{12})\,y_2 - 0.5\,\widehat{y}_{12_d}\,(y_{11}\,\mathsf{I}_3 - \widehat{y}_{12})\,y_2\,,\end{aligned} \tag{4.25}$$

where $\Lambda_3, \Lambda_4 > 0$ are gain matrices, $\widehat{J}^c_{0c}(x_1) = J^c_{0c}(x_1, \widehat{\Pi}_g)$.

2. *The stabilizing sliding surface is defined in* (4.22)*, and the corresponding inner control law is*

$$u(t) = \widehat{M}^0_{qq}(x_1)\left(w(t) - \Lambda_1 \lfloor s_x \rceil^{1/2} - \Lambda_2 \int_0^t \lfloor s_x \rceil^0 \, d\tau\right), \tag{4.26}$$

where $\Lambda_1, \Lambda_2 > 0$ *are gain matrices and* $\widehat{M}^0_{qq}(x_1) = M^0_{qq}(x_1, \widehat{\Pi}_g, \widehat{\Pi}_d)$.

Then, control laws (4.26) *and* (4.24) *ensure finite-time local convergence of the sliding variables* s_x, s_y *as defined in* (4.22) *and* (4.23)*. Furthermore, the errors* e_c*,* e_ω *are asymptotically stable under the dynamics of* $s_y = 0$.

The proof is presented on Section B.7, and it is based on finding appropriate bounds for the system disturbances. Figure 4.3 illustrates a block diagram for the proposed control scheme.

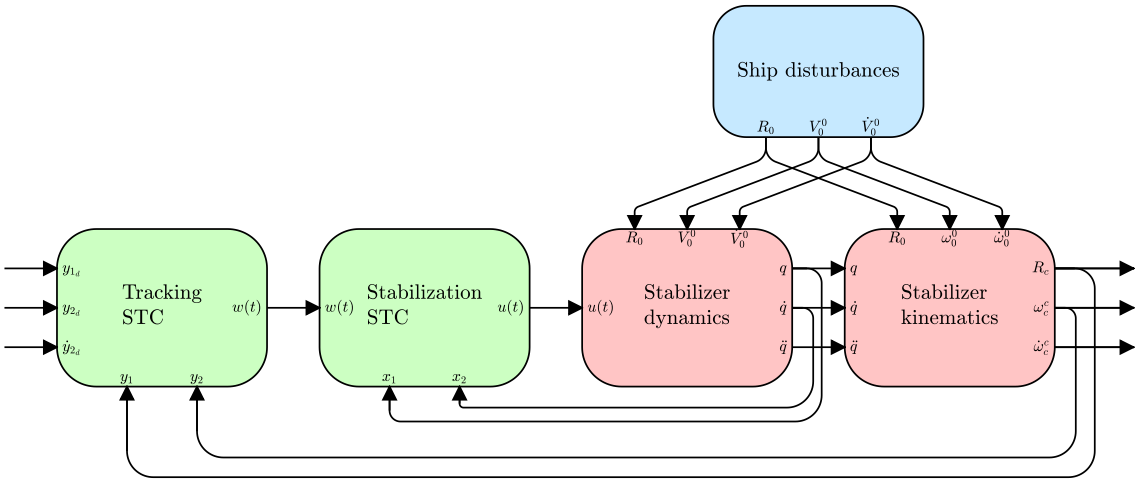


Figure 4.3: Block diagram for the closed-loop system with STC controller and full state feedback. The stabilization controller implements STC, using the joint angles q and velocities $\dot{q}$. The outer controller also implements STC, generating a continuous output and using the camera orientation r_c and angular velocity ω_c^c. Notice that, unlike the previous proposed controllers, no *feedback linearization* is employed.

4.2.2 Super-Twisting Control with HOSM Observer

If the joint velocity state $x_2 \in \mathbb{R}^3$ is not available, an observer could be used to estimate the joint velocity state $\widehat{x}_2(t)$ using measurements of $x_1(t)$. Because of its desired characteristics such as finite-time exact convergence, sliding mode observers could be used for this purpose, such as the *Super-Twisting* Observer (STO) [48]. However, as demonstrated in [47], it is not possible to achieve SOSM using *continuous* control when STC is implemented using Super-Twisting observers. A proposed

solution is to combine STC with a third-order HOSM observer to achieve *continuous* control on the joint torques $u(t)$ using output feedback.

Define the estimation errors

$$\begin{aligned} e_{x_1} &= x_1 - \hat{x}_1 \,, \\ e_{x_2} &= x_2 - \hat{x}_2 \,, \\ e_{x_3} &= x_3 - \hat{x}_3 \,, \end{aligned} \tag{4.27}$$

where $\hat{x}_1$, $\hat{x}_2$ and $\hat{x}_3$ are the system state estimates. A HOSM observer can be defined as the third-order system:

$$\begin{aligned} \dot{\widehat{x}}_1 &= K_1 \lfloor e_{x_1} \rceil^{2/3} + \widehat{x}_2 \,, \\ \dot{\widehat{x}}_2 &= K_2 \lfloor e_{x_1} \rceil^{1/3} + \widehat{x}_3 + (\widehat{M}_{qq}^0)^{-1} \, u \,, \\ \dot{\widehat{x}}_3 &= K_3 \lfloor e_{x_1} \rceil^{0} \,. \end{aligned} \tag{4.28}$$

where K_1, K_2 and K_3 are positive-definite matrices. From [49], the system (4.28) is finite-time stable over e_{x_1}, e_{x_1} and e_{x_1}, under some conditions on the gains. Then, after a finite time, $e_{x_2} = x_2 - \hat{x}_2 = 0$, which means that we can ideally estimate the joint velocities x_2 from measurements of the joint angles x_1 only. Then, one can define a *modified stabilizing* sliding variable as

$$\widehat{s}_x = \widehat{x}_2 - \int_0^t w(\tau) \, d\tau \,, \tag{4.29}$$

provided that SOSM can be achieved in $\widehat{s}_x$, leading to a similar result than in Theorem 7, but now using output feedback only. The *tracking* sliding variable is defined the same way as (4.23).

Remark 20. *Two HOSMOs could be designed: one for the joint velocities $x_2(t)$, and other for the camera angular velocity $y_2(t)$. However, usually the camera orientation $y_1(t)$ is obtained from an Inertial Measurement Unit (IMU), a device that combines measurements from gyroscopes (which measure angular velocity) and magnetometers (which measure magnetic fields), providing an (usually) accurate estimate for $y_1(t)$. Therefore, trustworthy direct measurements of $y_2(t)$ are usually already available.*

The following theorem provides an stability analysis for the proposed ST controller and HOSM observer.

Theorem 8 (*Cascade Super-Twisting Control with Output Feedback*)**.** *Let* (4.15) *and* (4.16) *be the dynamic and kinematic models of an ISP system under parametric uncertainties. Assume the following:*

(i) the body Jacobian $J^c_{0c_2} \in \mathbb{R}^{3\times 3}$ *is full-rank.*

(ii) the ISP joint velocities $x_2 \in \mathbb{R}^3$ *and joint accelerations* $\dot{x}_2 \in \mathbb{R}^3$ *are uniformly norm-bounded.*

(iii) the zero, first and second-order time-derivatives of the vehicle velocity twist $V_0^0 \in \mathbb{R}^6$ *are uniformly norm-bounded.*

The HOSM observer for the inner system (4.15) *is given by* (4.28) *with estimation errors defined in* (4.27)*. The tracking sliding variable and control law are defined as* (4.23) *and* (4.24)*, and the modified stabilizing sliding variable is given by* (4.29)*. The corresponding stabilizing control law is*

$$u(t) = \widehat{M}^0_{qq}(x_1)\left(w(t) - K_2\lfloor e_{x_1}\rceil^{1/3} - \Lambda_1\lfloor \widehat{s}_x\rceil^{1/2} - \Lambda_2\int_0^t \lfloor \widehat{s}_x\rceil^0\, d\tau\right). \tag{4.30}$$

Then, control laws (4.30) *and* (4.24) *with observer* (4.28) *ensure finite-time exact convergence of the sliding variables* $\widehat{s}_x, s_y$ *as defined in* (4.29) *and* (4.23)*, and of the estimation errors* (4.27)*. Furthermore, the errors* e_c, e_ω *are asymptotically stable under the dynamics of* $s_y = 0$*.*

The proof is presented on Section B.8, and it is also based on finding appropriate bounds for the disturbances. Figure 4.4 illustrates the proposed control scheme.

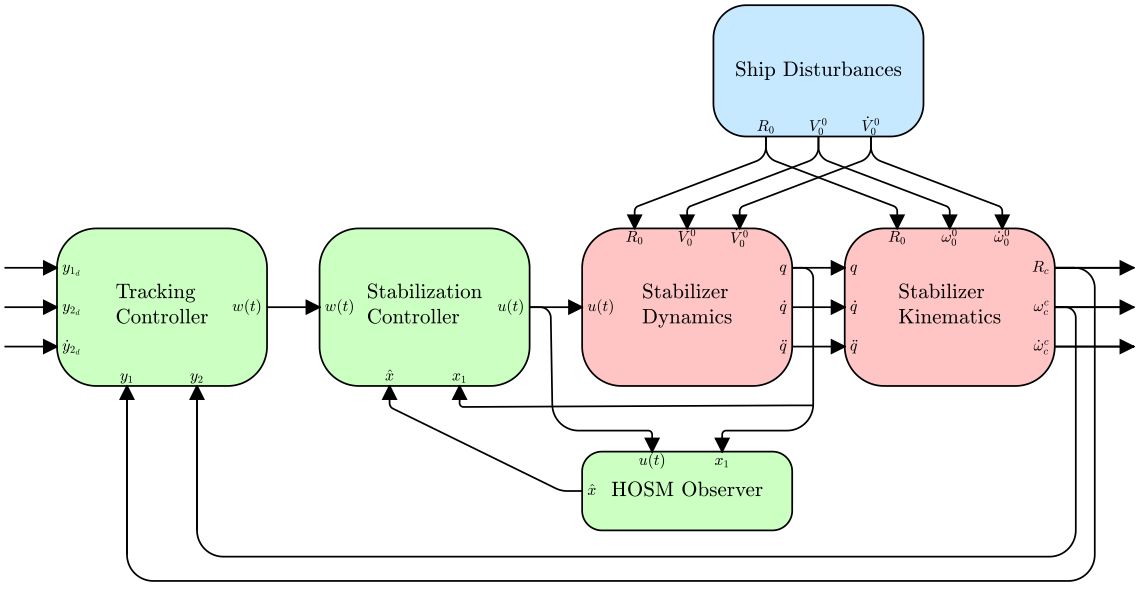


Figure 4.4: Block diagram for the closed-loop system with STC controller and output feedback using HOSMO.

Notice that, as before, the inner stabilization controller attempts to linearize the ISP dynamic system into an ideal double integrator, but using only measurements of the joint angles. The outer tracking controller is designed in the same way as

before. It is worth noticing that control law (4.30) only differs from (4.26) because of the term $K_2 \lfloor e_{x_1} \rceil^{1/3}$ and the use of $\widehat{s}_x$ instead of s_x.

4.3 Simulation Results

This section presents the results of the proposed control methods. MATLAB Simulink® models similar to the ones of Section 3.3 were implemented for the simulation of the dynamic model of the 3-DOF ISP installed on an arbitrary vehicle and for the implementation of the cascade Super Twisting Control (STC) strategies proposed in Section 4.2 (Fig. 4.5). Important modifications were made in the "ISP System" block on Fig. 3.5, so that the angular vehicle motion disturbances and internal orientation representation for the camera are expressed according to the quaternion formalism, instead of RPY angles.

The coefficients for the joint friction model (2.105) used in the simulations are shown in Table 4.1. A saturation of $\pm 12.2\, Nm$ in each joint motor was considered,

Table 4.1: Chosen coefficients for the joint friction model.

Parameter	$i=1$	$i=2$	$i=3$	Units
F_{brk_i}	0.0180	0.0150	0.0840	$N\,m$
F_{c_i}	0.0135	0.0113	0.0630	$N\,m$
F_{v_i}	0.0009	0.0008	0.0042	$N\,m\,s/rad$
ω_{brk_i}	0.0100	0.0100	0.0100	rad/s

and a 12 bit quantization was considered in the controller output. The joint encoders and the INS were modeled considering hardware effects such as resolution, bias, noise and quantization, and the base motion data (represented by variables r_0, V_0^0 and $\dot{V}_0^0$) were obtained from the same simulation data used in Section 3.3.

Remark 21. *The presented control methods can be applied to any kind of vehicle or moving base where the ISP is installed, since the quaternion formalism does not suffer from representation singularities and the base dynamics (velocities and accelerations) only affect the overall magnitude of the gains.*

The mass matrices in (4.26) and (4.30) and the Jacobian matrix in (4.24) were computed using numerical algorithms, implemented with MATLAB® mex files. The values for the nominal parameters used for the computation of $\widehat{M}_{qq}^0$ were once again set as the real values in Table 3.1 with a percentage of error. The parametric uncertainty on the ISP axes varies from 0° to 1° of misalignment between the real and

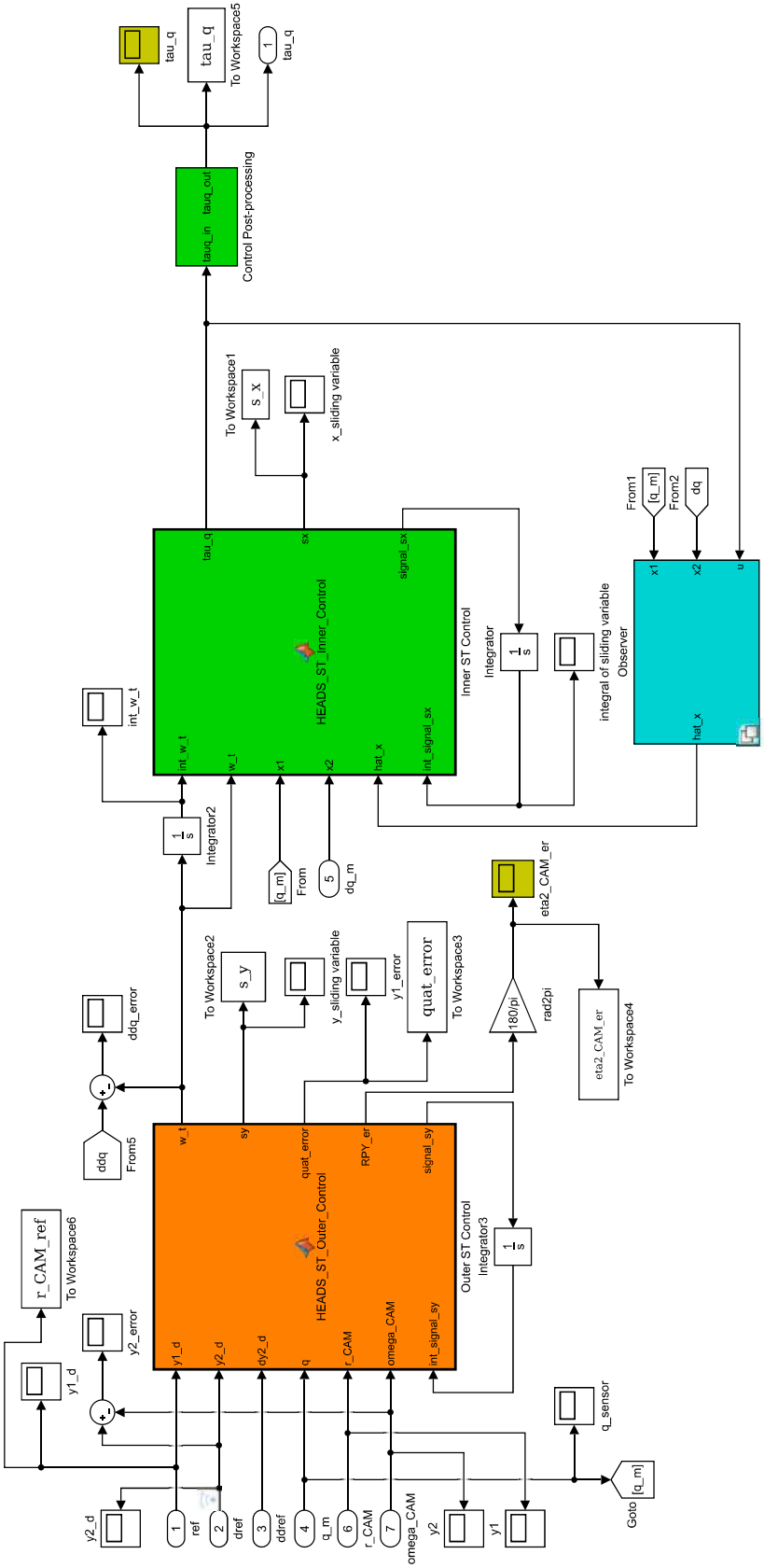


Figure 4.5: Block for simulation of the proposed STC controllers.

the nominal values. The gain matrices for both state and output feedback controllers were set as $K_c = 2\,\mathsf{I}_3\,rad\,s^{-1}$, $\Lambda_1 = \Lambda_3 = 5\,\mathsf{I}_3\,rad^{\frac{1}{2}}\,s^{\frac{3}{2}}$, $\Lambda_2 = \Lambda_4 = 5\,\mathsf{I}_3\,rad\,s^{-2}$ and the HOSMO gain matrices were chosen as $K_1 = 10\,\mathsf{I}_3\,rad^{\frac{1}{3}}\,s^{-1}$, $K_2 = 10\,\mathsf{I}_3\,rad^{\frac{2}{3}}\,s^{-2}$ and $K_3 = 10\,\mathsf{I}_3\,rad\,s^{-2}$. These values are sufficient to overcome the norm of the disturbances and small enough to reduce the control chattering.

The target point inertial reference p_t is given by (3.22), as before. The orientation and angular velocity/acceleration references y_{1_d}, y_{2_d}, $\dot{y}_{2_d}$ for the ISP are computed from the expression for the target position (3.22) and using the method on Section 2.2.4 for computing the desired LOS reference in quaternion space.

4.3.1 Full State Feedback STC

Initially, the results of the simulations for the STC with full state feedback are presented. The reference is given by trajectory (3.22) with a radius of $150\,m$, period of $T = 100\,s$ and $\epsilon = 1$. Due to the low translational velocity of the ship, this reference is equivalent to a fast target that is slowly circling the ship.

Figure 4.6 shows the transient and steady-state response of the state feedback STC in terms of RPY errors for the case of perfect knowledge of the ISP parameters. Both stabilization and tracking controllers achieve SOSM in *finite time* and in less than $1\,s$, with sliding precision on s_x and s_y approximately equal to $5 \times 10^{-4}\,mrad/s$, which is in accordance to the noise level of the simulated sensors. The quaternion error converges to the unit quaternion I_Q in less than $5\,s$. The resulting RPY jitter is limited to a small region of $0.03°$, due to measurement noise. The control signal is *continuous* and limited to the actuator saturation limits. However, it carries a high frequency component of approximately $60\,Hz$ due to the characteristic discontinuity in the time derivative of the Super-Twisting controller. The high frequency torque component reaches a maximum of $1\,Nm$ of amplitude at the pitch joint, which is definitively implementable by high quality drivers. The internal control signal w from the tracking STC controller remains bounded by $\|w\| < 0.15\,rad\,s^{-1}$.

Note that the full state feedback STC was able to reject all *unknown* nonlinear disturbances due to the ISP dynamics, kinematics and joint friction, with a precision ultimately bounded by the precision of the sensors and a fast and well behaved transient response. Note also that all these terms enter the dynamic equation as disturbances that are *matched* to the control variable. It is worth saying that, in the same way as in Section 3.3.2 and Section 3.3.3, the RPY error shown in Fig. 4.6 is the real RPY error constructed from the inertial sensors and GPS measurements. Therefore, in practice, there will exist the same kind of *offset* that was observed in the results of Section 3.3.4. Since this offset is due to the method for computing the

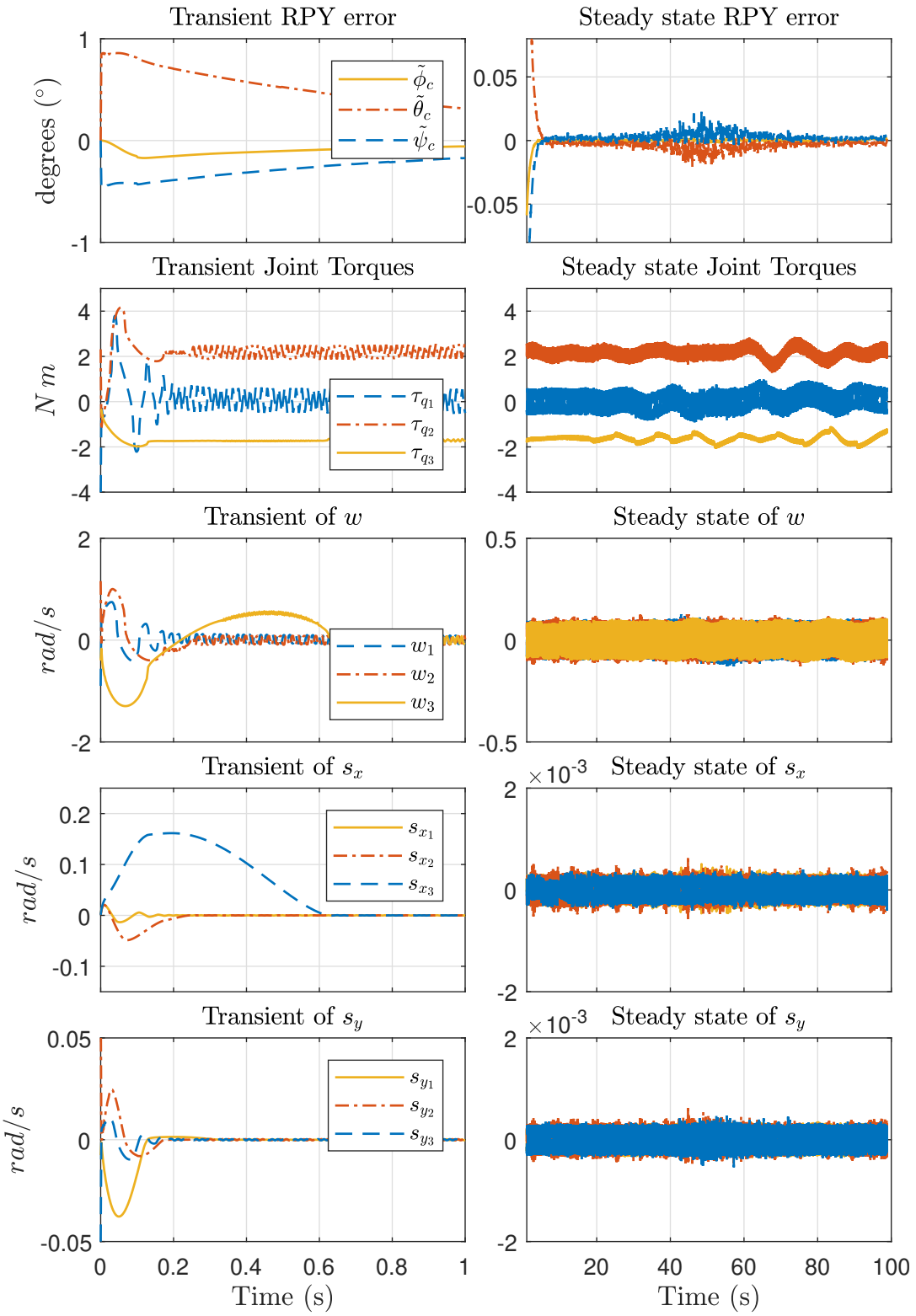


Figure 4.6: Response for state feedback STC controller with perfect knowledge of the ISP parameters.

LOS orientation reference, it cannot be eliminated by control.

Since an *estimated* mass matrix is used by the STC controller and due to the perfect parameter knowledge, the ISP dynamic system was successfully decoupled

into a double integrator by the stabilization controller. Besides, the RPY jtter is similar to the one obtained with the CT-PID methods, under full knowledge of the ISP parameters. Next, we are going to evaluate if the the proposed STC is also robust against parametric uncertainty, as stated by Theorem 7.

Several simulations where made under the same conditions, but with higher parametric uncertainty. Figure 4.7 shows the results obtained for the case of 50% of parametric error and 0.5° of error on all three ISP axis. Notice how the RPY error amplitude is unaffected by a large amount of parametric uncertainty. The joint torques and the inner control signal w are also very similar to the previous case, with a small decrease in the amplitude of the joint torque jitter. It can be explained by a decrease in the eigenvalues of the mass matrix $\widehat{M}_{qq}^{0}$ in (4.26), due to the 50% of error in the dynamic parameters. The sliding precision and reaching time were affected, since the parametric uncertainty introduces larger terms into the overall disturbance that the STC must reject (see Section B.7).

Above 60% of parametric error, the closed-loop system starts to reach its stability margin, as it is possible to observe divergence of the sliding variables from their sliding modes. However, by increasing the STC gains, it is possible to improve the controller stability margin, at the price of increasing the amplitude of the high frequency component in the control signals. Therefore, the value chosen for the STC gains Λ_1, Λ_2, Λ_3 and Λ_4 guarantees a satisfactory disturbance rejection and LOS tracking, even for up 60% of parametric uncertainty and up to 1° of axis misalignment, which is much higher than the common uncertainty value associated to most ISP designs. Note that the observed RPY jitter for the STC is similar to the CT-PID result with full knowledge of the system parameters, which is an experimental validation of the matching between the STC *equivalent control* and the *feedback linearization* control.

Next, we consider the effect of a first-order linear actuator dynamics with a settling time of $25\,ms$, which is a realistic value for the response time of a electronic driver. Considering the settling time as 4 times the time constant, the equivalent actuator transfer function has a distant pole at $160\,rad/s$. Figure 4.8 shows the response for the case of 50% of parametric error and the presence of the driver dynamics. The performance is strongly affected in the presence of unmodeled dynamics, with sliding precision approximately 10 times worse than in Fig. 4.7 for both s_x and s_y. The torque chattering is much higher as well, with chattering period of approximately $0.16\,s$ and $4\,Nm$ of amplitude in the yaw joint motor.

Clearly, under the presence of the actuator dynamics, *ideal* sliding is lost, and the sliding variables converge to a small bound. These bounds are smaller for faster unmodeled dynamics, reaching zero for infinitely fast dynamics. It means that, in

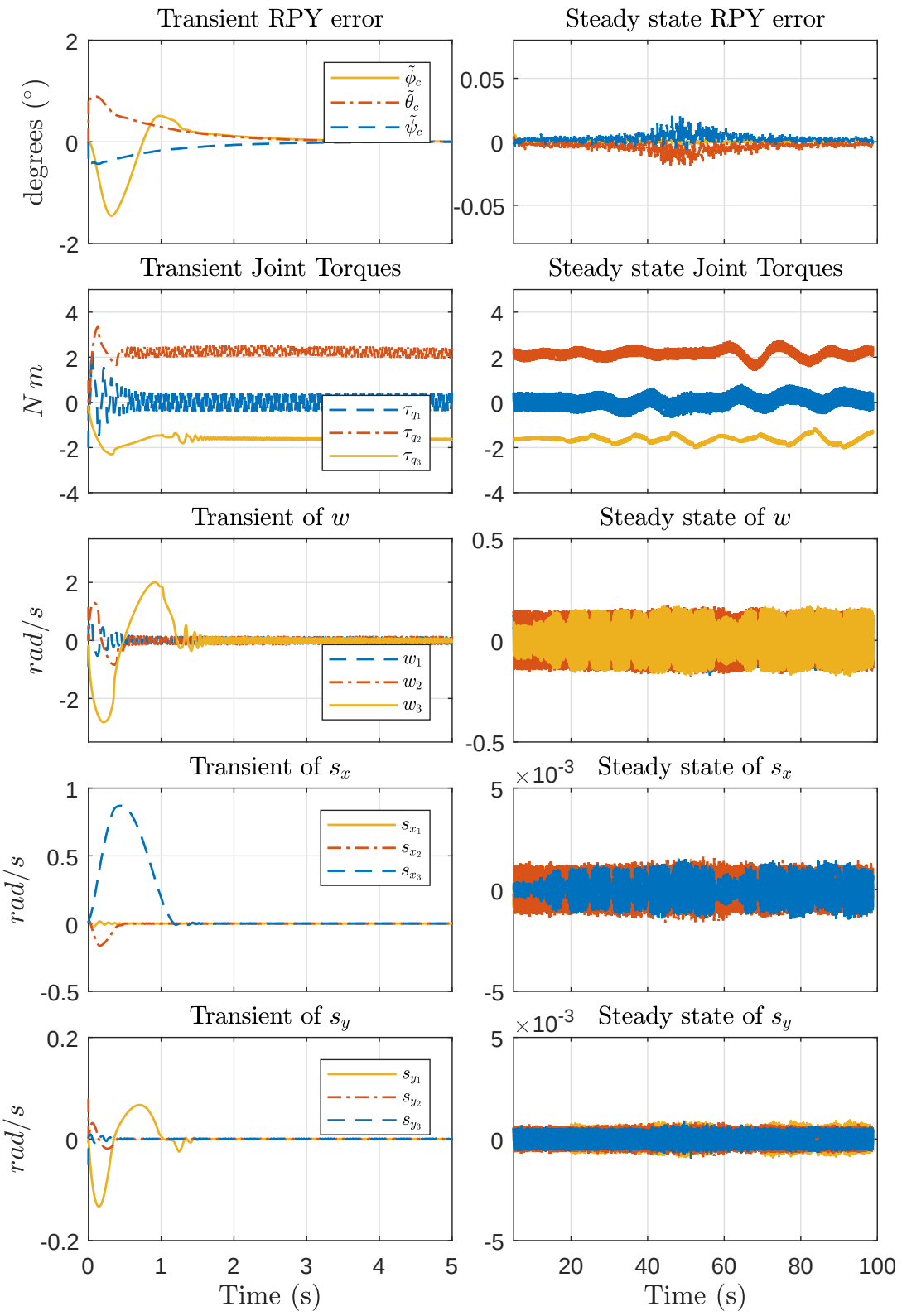


Figure 4.7: Response for state feedback STC controller with 50% of parametric error and 0.5° of axis error.

practice, the sliding precision is limited by the response time of the driver.

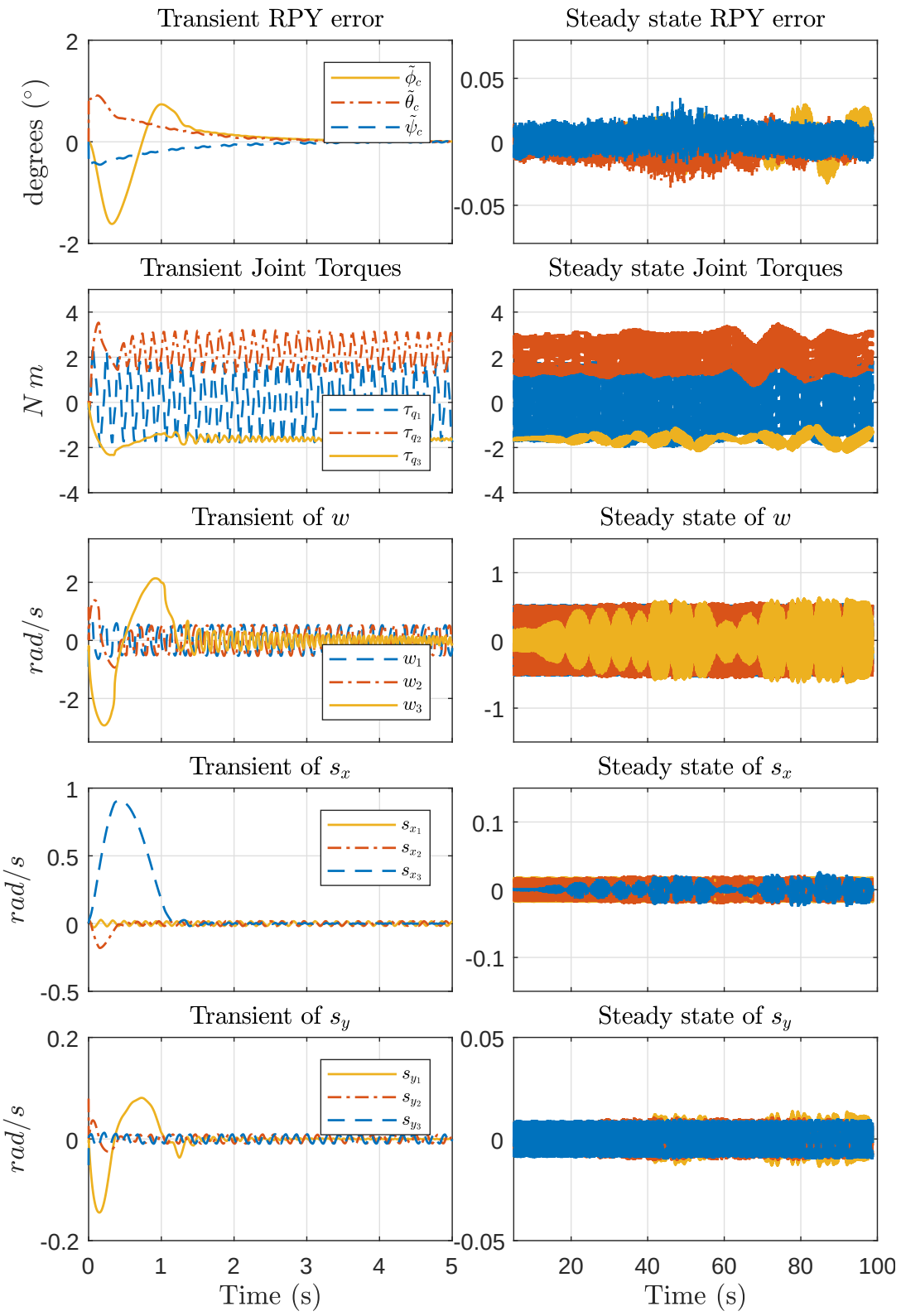


Figure 4.8: Response for state feedback STC controller with 50% of parametric error, 0.5° of axis error and first-order driver dynamics.

4.3.2 Output Feedback STC + HOSMO

Combining the stabilizer STC with the HOSM observer into an output feedback STC has the advantage of using joint angle measurements only, which can be many times more accurate than conventional joint velocity measurements. Here, the results for the output feedback STC scheme are presented. Its performance is going to be compared to the previously presented state feedback STC.

Figure 4.9 shows the transient and steady state response of the of the output feedback STC, again supposing perfect knowledge of the ISP parameters. The transient and performance of the RPY error remains practically the same than in Fig. 4.6. The amplitude of the control chattering suffered an increase due to the presence of the term multiplying K_2 in (4.30). The sliding precision of the stabilization sliding variable $\hat{s}_x$ is much better now, since the estimated joint velocity state is used in (4.29), instead of the actual joint velocity state x_2. The reaching time for $\hat{s}_x$ is the same as in Fig. 4.6. On the other hand, the sliding precision and reaching time of the tracking sliding variable $\hat{s}_y$ were slightly affected.

The HOSMO estimation errors for the output feedback STC scheme are shown in Fig. 4.10. A sliding mode is observed in e_{x_1}, e_{x_2} and e_{x_3}. All three estimation errors reach their sliding modes in approximately $0.6\,s$, and their norms converge to small regions of $2 \times 10^{-4}\,rad$, $3 \times 10^{-3}\,rad$ and $0.2\,rad$, respectively.

Next, we proceed using the same strategy as before, and testing the robustness of the output feedback STC against parametric uncertainty. Again, simulations where made under the same conditions, but with a rising parametric uncertainty. For up to 50% of parametric uncertainty, the closed-loop system was able to reject the matched disturbances just as before. However, the inclusion the the HOSM observer seems to have affected the stability margin of the closed-loop system.

Figure 4.11 illustrates the results for 50% of parametric uncertainty. The RPY error jitter is completely unaffected, as expected, and the amplitude of the joint torques is in fact smaller, which initially can be though of as an unexpected result. However, this effect is related to the eigenvalues of the mass matrix $\widehat{M}^0_{qq}(x_1)$, with can be interpreted as a variable gain that is dependent on the ISP nominal parameters. The 50% variation on the ISP nominal parameters with respect to the real ones can be positive or negative. In our simulations, we are generally considering a negative variation; however, for a positive 50% variation, the jitter amplitude on the joint torques is actually higher than in Fig. 4.10. Therefore, the increase or decrease of the control jitter is dependent on the direction of variation of the disturbance. This is also true for the state feedback STC.

Figure 4.12 illustrates the convergence of the estimation errors. For 50% of

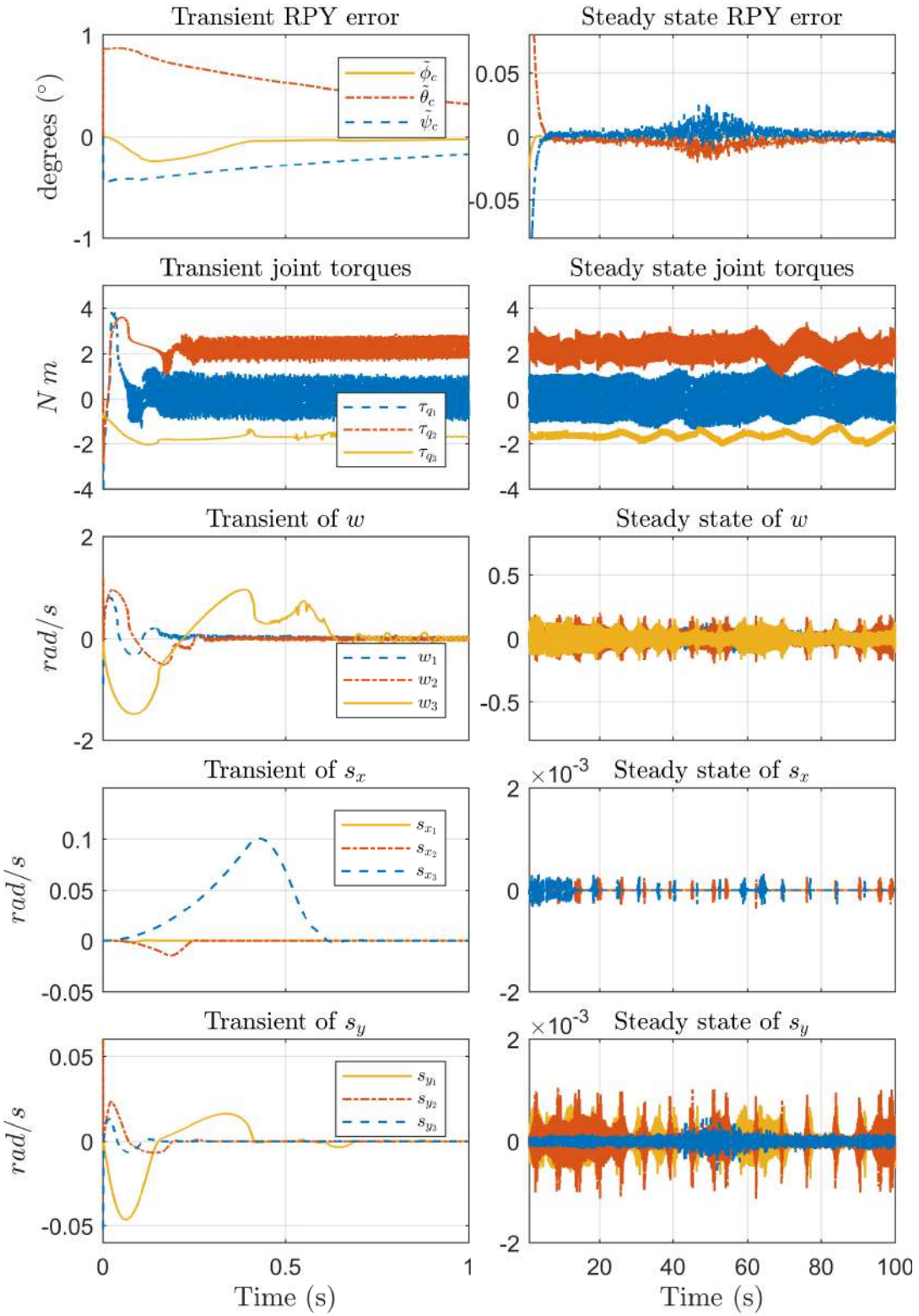


Figure 4.9: Response for the output feedback STC controller with perfect knowledge of the ISP parameters.

parameter uncertainty, the reaching time increases to approximately 3 s for e_{x_1}, e_{x_2} and e_{x_3}. The sliding precision does dot change for e_{x_1} and e_{x_2}, but it is slightly worse for e_{x_3}. This can be explained by Remark 26 in Section B.8. In the presence

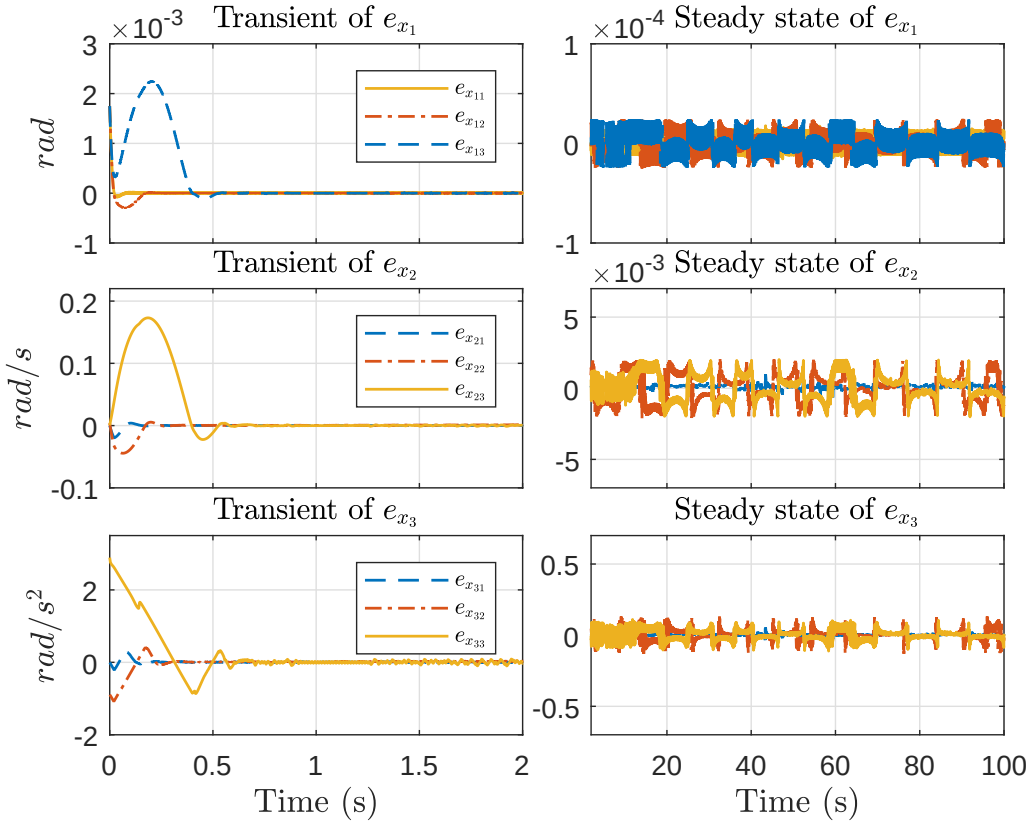


Figure 4.10: HOSMO estimation errors for the output feedback STC controller with perfect knowledge of the ISP parameters.

of parametric uncertainty, e_{x_3} does not converge to zero, but to the residue

$$\beta(\widetilde{\Pi}) = \left({M_{qq}^0}^{-1} - \widehat{M_{qq}^0}^{-1} \right) u \,.$$

Above 50% of parameter uncertainty, we begin to notice signs of divergence on the sliding variables and on the estimation errors. Therefore, we conclude that the robustness of the output feedback STC with respect to parameter uncertainty is slightly inferior to the state feedback STC, for the same control gains and the same dynamic disturbance. Comparing (B.38) with (B.48), we notice that $\overline{d}_y$ has an additional term when compared to d_y. Therefore, the $\left\|\overline{d}_y\right\| > \|d_y\|$, which means that Λ_3 and Λ_4 (the gains of the tracking STC) must be higher for the output feedback STC, in order to reject the same amount of parametric disturbance.

Lastly, we consider the effect of the same first-order linear actuator dynamics with $25\,ms$ of rising time considered in Section 4.3.1. The results are shown in Figs. 4.13 and 4.14. Once again, performance is affected in the presence of unmodeled dynamics, but the resulting RPY jitter is in fact better than the state feedback STC under the same conditions. The amplitude of the control jitter is also smaller, and as before, the sliding precision is better, regardless of the presence of the ac-

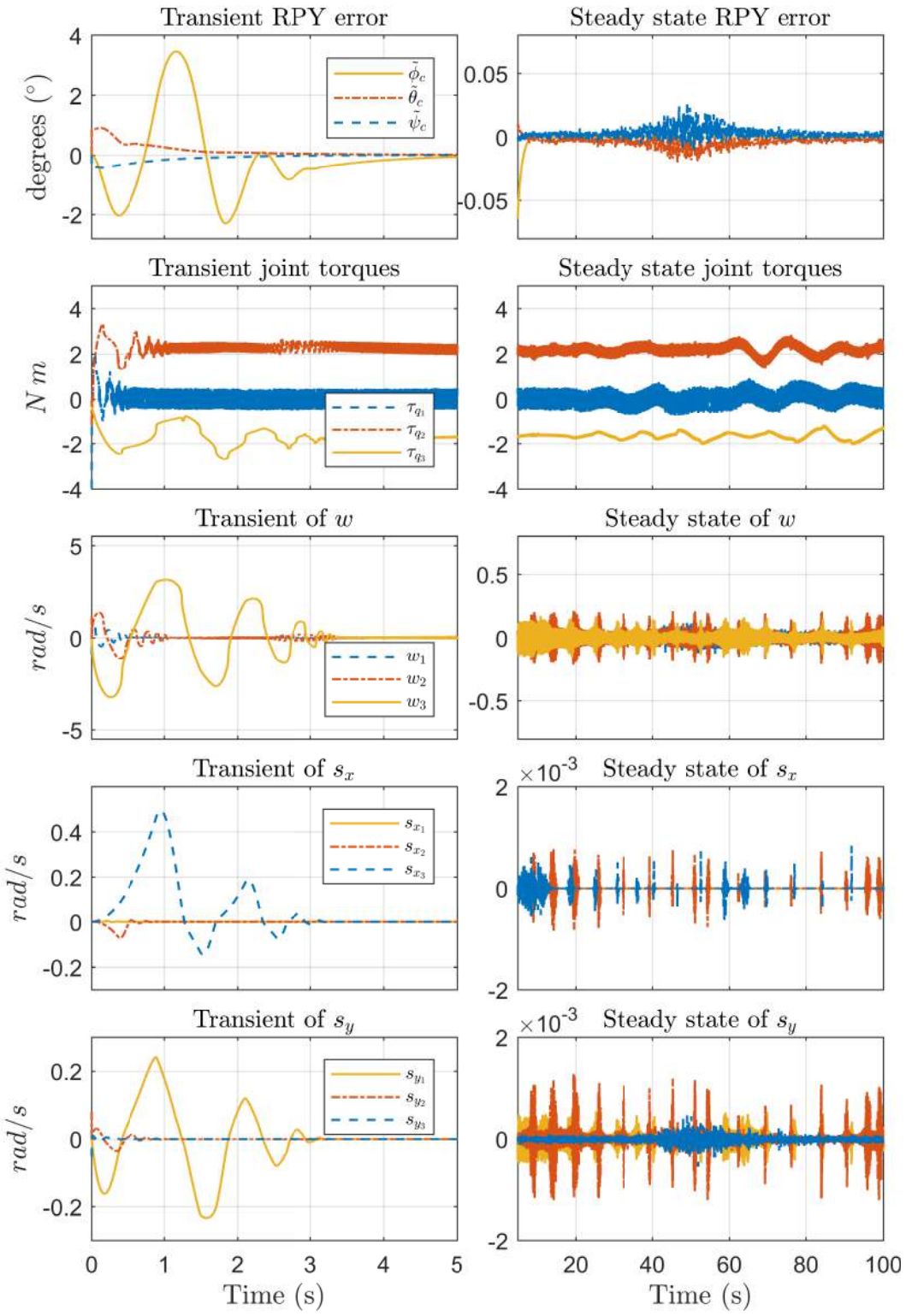


Figure 4.11: Response for the output feedback STC controller with 50% of parametric error and 0.5° of axis error.

tuator dynamics. Since the control signal is partially canceled out in the HOSM observer due to term $\widehat{M}_{qq}^{0}\,u$ in the second equation of (4.28), the presence of the actuator dynamics affects the dynamics of the stabilization controller much more than

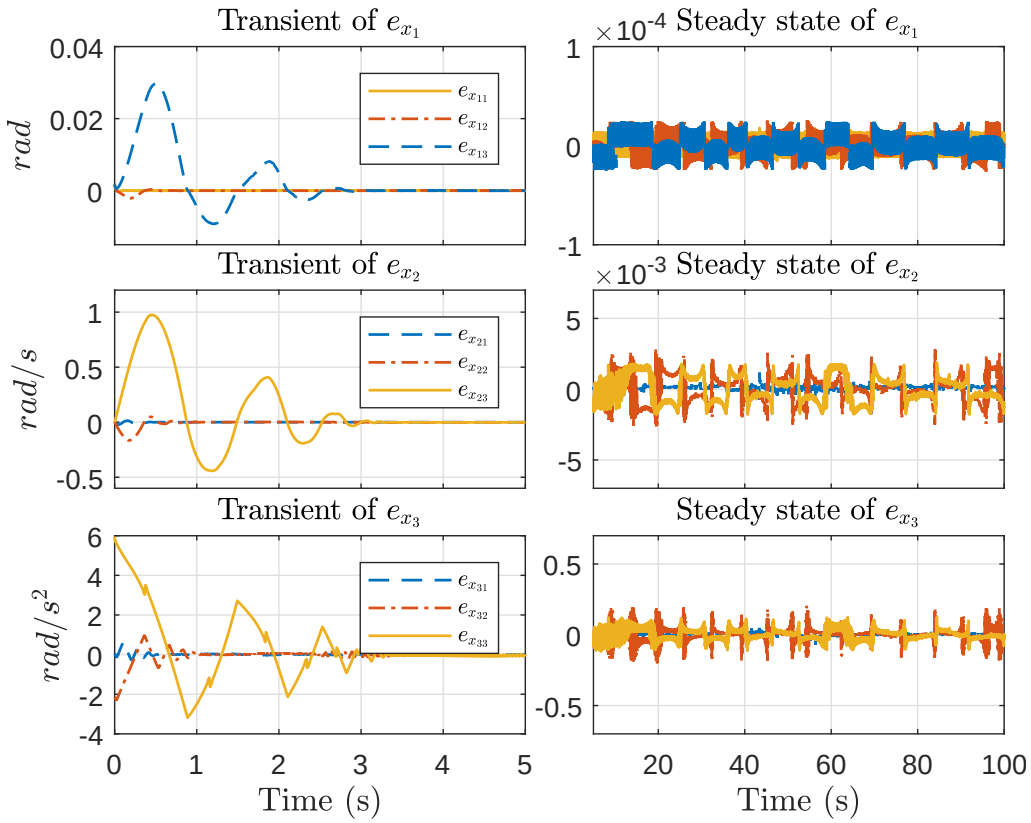


Figure 4.12: HOSMO estimation errors for the output feedback STC controller with 50% of parametric error and 0.5° of axis error.

the observer dynamics. The reaching times for s_x and s_y were also affected, rising up to $4\,s$, and the stability margin of the closed-loop system is impacted. However, the sliding precision on s_x and s_y is slightly better then in the state feedback STC result, shown in Fig. 4.8.

In Fig. 4.14, note that finite-time convergence of the estimation errors is achieved in approximately $4\,s$, the same as for the sliding variables of the controller. The sliding error precision and transient are worse then in Fig. 4.12, but still acceptable. Clearly, in terms of RPY error jitter and joint torque amplitudes, the presence of the unmodeled dynamics had a greater impact on the performance of the state feedback STC than on the output feedback STC.

Finally, Fig. 4.15 illustrates a comparison between the joint torque disturbances that compose the ideal equivalent control (4.17) and the actual control signal, after both signals were filtered using a a lowpass finite impulse response (FIR) filter with normalized passband frequency of $2\pi\,rad/s$, stopband frequency of $3\pi\,rad/s$, passband ripple of 0.5 dB, and stopband attenuation of 65 dB. As stated, for all shown cases, the equivalent control is able to match all unknown disturbances acting on on the ISP dynamics.

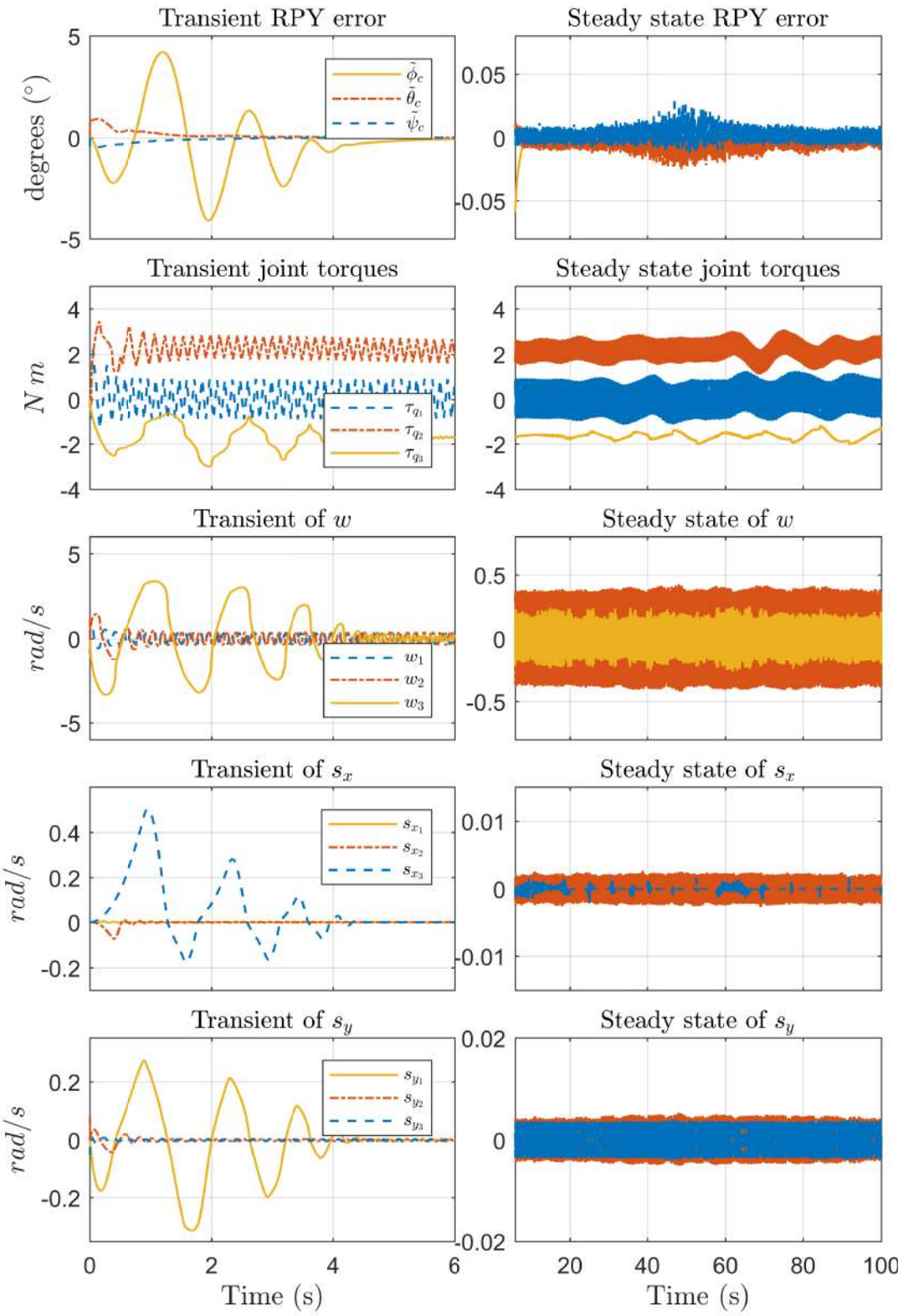


Figure 4.13: Response for the output feedback STC controller with 50% of parametric error, 0.5° of axis error and first-order driver dynamics.

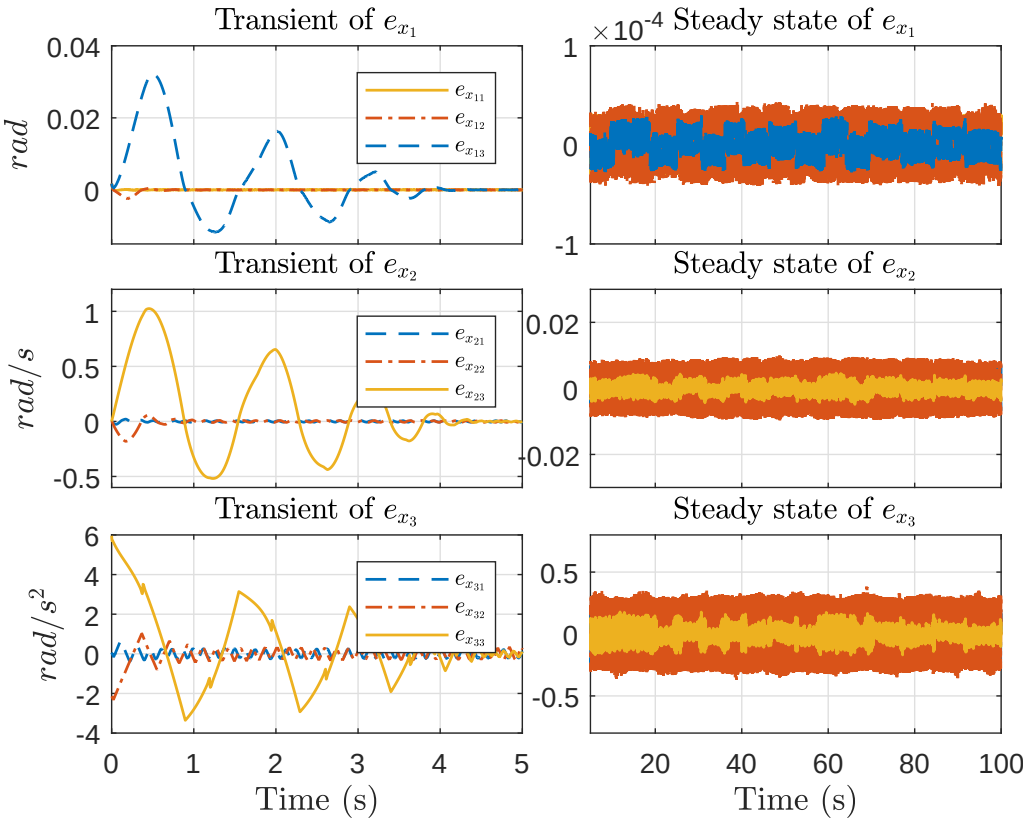


Figure 4.14: HOSMO estimation errors for the output feedback STC controller with 50% of parametric error, 0.5° of axis error and first-order driver dynamics.

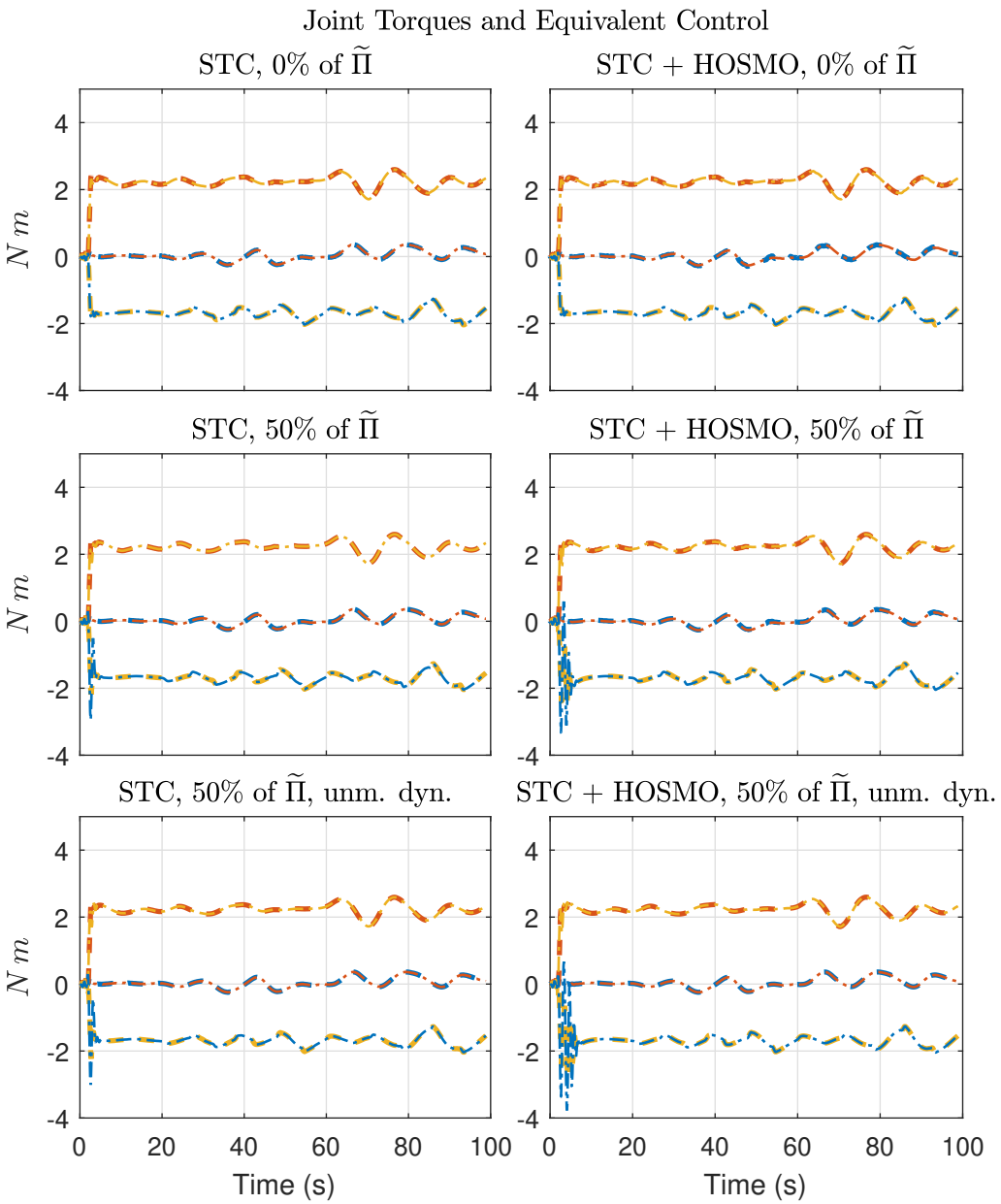


Figure 4.15: Comparison between equivalent control and joint torque disturbances.

Chapter 5

Conclusion and Future Works

In this work, the problem of LOS stabilization and tracking using inertial stabilization platforms was tackled. We have used the VMS framework to obtain a realistic mechanical model of an ISP mounted on a vehicle, considering design imperfections such as mechanical unbalance and non-orthogonal gimbal axes.

First, two CT-PID controllers in both direct and indirect configurations were proposed and their performance was compared to the P-PI controller, which is usually employed for LOS stabilization applications. Ideally, the proposed controllers guarantee exponential stability on the closed-loop system. In practice, however, the error amplitude is bounded by a small residual set which is linear in the parametric error. Their effectiveness was demonstrated by means of realistic simulations.

The results suggest better *precision* than the P-PI controller even in the presence of significant parameter uncertainty, up to 35% in the ISP parameters, considering the applied reference trajectories. In terms of pointing *accuracy*, the performance of the direct CT-PID scheme is equivalent to the P-PI controller, due to the direct measurement of the camera motion using the directly positioned INS. However, since the indirect CT-PID scheme relies on the perfect knowledge of the ISP geometric parameters to compute the camera position, its pointing accuracy is strongly dependent on the kinematic model of the ISP. In the direct case, the pointing accuracy is limited to the accuracy of the GPS positioning system. This introduces a practical minimal offset in the RPY errors that cannot be mitigated by control, since it is due to errors in the computation of the LOS reference.

These controllers have the disadvantage of using a complex mathematical model instead of a simple linear filter, which is the case of the P-PI controller. However, they do provide commercial stabilization performance, specially when the ISP parameters are well known. This could be the case if identification techniques are employed during the design phase of the ISP.

To improve performance even further, two sliding mode controllers were proposed to overcome the limitations of the CT-PID methods. First, the LOS tracking problem was formulated by means of the unit quaternion formalism, to avoid representation singularities. Second, the proposed controllers provide finite time stabilization and ideal disturbance rejection even in the presence of parametric uncertainty. According to the sliding mode control literature, it means that the super twisting controllers are able to generate an *equivalent control* signal that perfectly matches the dynamic and kinematic disturbances of the system.

Some important remarks can be summarized.

1. Both output and state feedback ST controllers are able to reject all matched disturbances on the dynamic and kinematic ISP equations, guaranteeing stabilization of the ISP and tracking of the LOS reference with the same performance of the CT-PID controller with full knowledge of the system parameters.

2. The output feedback STC guarantees the same performance on the RPY error rejection than the state feedback STC, but it tends to produce a slightly higher control jitter due to the term $K_2 \lfloor e_{x_1} \rceil^{1/3}$ on the control law (4.30).

3. Although the sliding precision on the estimation errors e_{x_1} and e_{x_2} is not affected by the presence of parametric uncertainty, the accuracy on e_{x_3} is affected, according to Remark 26 in Section B.8.

4. The presence of unmodeled actuator dynamics greatly influences the closed-loop response for both controllers, affecting the RPY error and control jitter, reaching time of the sliding and estimation variables and also the stability margin of the whole system.

The results obtained in Section 4.3 and the above conclusions suggest the proposed STC controllers as a feasible solution for implementation on a real system, achieving a *minimum* level of precision of $0.05°$ or approximately $0.8\,mrad$ under the presence of relatively slow actuator dynamics. Considering that high-performance actuators have a much faster response time than $25\,ms$ used in the simulations, these controllers could achieve a precision on the level of $100\mu rad$ if precise sensors are used, which places them at the high-precision LOS stabilization category.

5.1 Future Works

Some important future works and developments for this research are:

1. to formulate the camera orientation in terms of unit quaternions in the computed-torque schemes, allowing them to be applied to vehicles with larger motion amplitude than a vessel.

2. to implement the proposed controllers in an experimental platform and perform practical tests, validating the effectiveness of the proposed methods by means of experimental data.

3. to investigate how *adaptation* on the gains can be used to tackle the problem of unknown disturbance bounds, following the trend of [50], [51].

4. to investigate how the undesirable effects of the motor unmodeled dynamics can be mitigated by using *voltage control* instead of torque control.

5. to investigate how *higher-order* sliding mode controllers could be used to tackle the LOS stabilization and tracking problem.

5.2 Publications

This work has produced four accepted papers in international conferences:

1. "Dynamic Model and Line of Sight Control of a 3-DOF Inertial Stabilization Platform" [52]. Authors: Matheus F. Reis, Guilherme P. S. Carvalho, Alex F. Neves, Alessandro J. Peixoto. Presented at 2018 American Control Conferece, at Milwaukee, USA.

2. "Identificação e Controle por Torque Computado de uma Plataforma Inercial para Estabilização e Rastreamento da Linha de Visada". Authors: Matheus F. Reis, João C. Monteiro, Guilherme P. S. Carvalho, Alex F. Neves, Alessandro J. Peixoto. Presented at the XXII Congresso Brasileiro de Automática (CBA2018).

3. "Super-Twisting Control with Quaternion Feedback for Line-of-Sight Stabilization and Tracking". Authors: Matheus F. Reis, Ramon R. Costa, Antonio C. Leite. Presented at the XXII Congresso Brasileiro de Automática (CBA2018).

4. "Super-Twisting Control with Quaternion Feedback for a 3-DOF Inertial Stabilization Platform". Authors: Matheus F. Reis, João C. Monteiro, Ramon R. Costa, Antonio C. Leite. Accepted for publication at the 57th IEEE Conference on Decision and Control (CDC2018).

Bibliography

[1] HILKERT, J. M. "Inertially stabilized platform technology: Concepts and principles", *IEEE Control Systems Magazine*, v. 28, n. 1, pp. 26–46, 2008. doi: 10.1109/MCS.2007.910256. Disponível em: <https://ieeexplore.ieee.org/document/4430218/>.

[2] MASTEN, M. K. "Inertially stabilized platforms for optical imaging systems", *IEEE Control Systems Magazine*, v. 28, n. 1, pp. 47–64, Feb. 2008. doi: 10.1109/MCS.2007.910201.

[3] FRANGIPANI, V. *Estabilização Inercial De Câmera Pan-Tilt De Um ROV*. Master dissertation, Federal University of Rio de Janeiro/COPPE, Department of Electrical Engineering, 2013.

[4] DEBRUIN, J. "Control systems for mobile Satcom antennas", *IEEE Control Systems*, v. 28, pp. 86–101, 2008.

[5] ABDO, M., VALI, A. R., TOLOEI, A., et al. "Research on the cross-coupling of a two axes gimbal system with dynamic unbalance", *International Journal of Advanced Robotic Systems*, v. 10, 2013. ISSN: 17298806. doi: 10.5772/56963.

[6] HURÁK, Z., REZÁČ, M. "Image-based pointing and tracking for inertially stabilized airborne camera platform", *IEEE Transactions on Control Systems Technology*, v. 20, n. 5, pp. 1146–1159, 2012. ISSN: 10636536. doi: 10.1109/TCST.2011.2164541.

[7] ABDO, M. M., VALI, A. R., TOLOEI, A. R., et al. "Stabilization loop of a two axes gimbal system using self-tuning PID type fuzzy controller", *ISA Transactions*, v. 53, n. 2, pp. 591 – 602, 2014. ISSN: 0019-0578. doi: https://doi.org/10.1016/j.isatra.2013.12.008. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0019057813002188>.

[8] KHALIL, H. K. *Nonlinear systems; 3rd ed.* Upper Saddle River, NJ, Prentice-Hall, 2002. Disponível em: <https://cds.cern.ch/record/1173048>.

[9] SHTESSEL, Y., EDWARDS, C., FRIDMAN, L., et al. *Sliding Mode Control and Observation*. Springer, 2014. ISBN: 1489991220, 9781489991225.

[10] MORISITA, M., YAGI, T. "The stability of human eye orientation during visual fixation and imagined fixation in three dimensions", *Auris Nasus Larynx*, v. 28, n. 4, pp. 301–304, Nov. 2001.

[11] VANNUCCI, L., FALOTICO, E., TOLU, S., et al. "Eye-Head Stabilization Mechanism for a Humanoid Robot Tested on Human Inertial Data". In: *Biomimetic and Biohybrid Systems: 5th International Conference, Living Machines 2016, Edinburgh, UK, July 19-22, 2016. Proceedings*, pp. 341–352, Cham, Springer International Publishing, 2016. ISBN: 978-3-319-42417-0. doi: 10.1007/978-3-319-42417-0_31. Disponível em: <https://doi.org/10.1007/978-3-319-42417-0_31>.

[12] WINDAU, J., ITTI, L. "Multilayer real-time video image stabilization". In: *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2397–2402, Sep. 2011. doi: 10.1109/IROS.2011.6094738.

[13] J. M. HILKERT, GAVIN KANGA, K. K. "Line-of-sight kinematics and corrections for fast-steering mirrors used in precision pointing and tracking systems". In: *Proceedings of SPIE - The International Society for Optical Engineering*, v. 9076, pp. 9076 – 9076 – 15, 2014. doi: 10.1117/12.2049857. Disponível em: <https://doi.org/10.1117/12.2049857>.

[14] KENNEDY, P. J., KENNEDY, R. *Stabilizing the Line of Sight*. Peter J Kennedy, 2014. ISBN: 0692273565, 9780692273562.

[15] KENNEDY, P. J., KENNEDY, R. L. "Direct versus indirect line of sight (LOS) stabilization", *IEEE Transactions on Control Systems Technology*, v. 11, n. 1, pp. 3–15, 2003. doi: 10.1109/TCST.2002.806443.

[16] BIGLEY, W. J., SCHUPAN, F. "Wideband Base Motion Isolation Control for a Mobile Platform", *Proceedings of the American Control Conference*, pp. 1483–1490, 1987. ISSN: 07431619.

[17] BATTISTEL, A., LIZARRALDE, F., HSU, L. "Estabilização de Plataformas Inerciais Utilizando Apenas duas Medidas de Sensores Giroscópicos", *Congresso Brasileiro de Automática*, pp. 215–222, Set. 2010.

[18] FROM, P. J., DUINDAM, V., GRAVDAHL, J. T., et al. "Modeling and motion planning for mechanisms on a non-inertial base". In: *2009 IEEE International Conference on Robotics and Automation*, pp. 3320–3326, May 2009. doi: 10.1109/ROBOT.2009.5152666.

[19] GRAVDAHL, J. T., FROM, P. J., PETTERSEN, K. Y. *Vehicle-Manipulator Systems - Modeling for Simulation, Analysis, and Control*. 1st ed. , Springer-Verlag London, Jan. 2014. ISBN: 9781447154624. doi: 10.1007/978-1-4471-5463-1.

[20] SICILIANO, B., SCIAVICCO, L., VILLANI, L. *Robotics : Modelling, Planning and Control*. Advanced Textbooks in Control and Signal Processing. London, Springer, 2009. ISBN: 978-1-84628-641-4. doi: 10.1007/978-1-84628-642-1. Disponível em: <https://www.springer.com/us/book/9781846286414>.

[21] MURRAY, R. M., SASTRY, S. S., ZEXIANG, L. *A Mathematical Introduction to Robotic Manipulation*. 1st ed. , CRC Press, Inc., 1994. ISBN: 0849379814. Disponível em: <https://www.cds.caltech.edu/~murray/books/MLS/pdf/mls94-complete.pdf>.

[22] HURÁK, Z., ŘEZÁČ, M. "Control design for image tracking with an inertially stabilized airborne camera platform". In: *Proceedings SPIE - The International Society for Optical Engineering*, v. 7696, pp. 7696 – 7696 – 12, 2010. doi: 10.1117/12.849902. Disponível em: <http://dx.doi.org/10.1117/12.849902>.

[23] ŘEZÁČ, M., HURÁK, Z. "Vibration rejection for inertially stabilized double gimbal platform using acceleration feedforward", *Proceedings of the IEEE International Conference on Control Applications*, pp. 363–368, 2011. ISSN: 1085-1992. doi: 10.1109/CCA.2011.6044442.

[24] PIATKOWSKI, T. "Dahl and LuGre dynamic friction models: The analysis of selected properties", *Mechanism and Machine Theory*, v. 73, n. Supplement C, pp. 91 – 100, 2014. ISSN: 0094-114X. doi: https://doi.org/10.1016/j.mechmachtheory.2013.10.009. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0094114X13002152>.

[25] REZAC, M. *Inertial stabilization, estimation and visual servoing for aerial surveillance*. Doctoral thesis, Czech Technical University in Prague, 2013. Disponível em: <https://support.dce.felk.cvut.cz/mediawiki/images/e/e7/Diz_2013_rezac_martin.pdf>.

[26] KÖNIGSEDER, F., KEMMETMÜLLER, W., KUGI, A. "Attitude control strategy for a camera stabilization platform", *Mechatronics*, v. 46, n. Supplement C, pp. 60 – 69, July 2017. ISSN: 0957-4158. doi: 10.1016/j.mechatronics.2017.06.012. Disponível em: <https://doi.org/10.1016/j.mechatronics.2017.06.012>.

[27] WRONKA, C., DUNNIGAN, M. "Derivation and analysis of a dynamic model of a robotic manipulator on a moving base", *Robotics and Autonomous Systems*, v. 59, n. 10, pp. 758 – 769, 2011. ISSN: 0921-8890. doi: 10.1016/j.robot.2011.05.010. Disponível em: <https://doi.org/10.1016/j.robot.2011.05.010>.

[28] KÖNIGSEDER, F., KEMMETMÜLLER, W., KUGI, A. "Attitude Estimation Using Redundant Inertial Measurement Units for the Control of a Camera Stabilization Platform", *IEEE Transactions on Control Systems Technology*, v. 24, n. 5, pp. 1837–1844, Jan. 2016. ISSN: 1063-6536. doi: 10.1109/TCST.2015.2510324.

[29] FADALI, M. S., YAZ, E. "Stability robustness and robustification of the exact linearization method of robotic manipulator control". In: *Proceedings of 1995 34th IEEE Conference on Decision and Control*, v. 2, pp. 1624–1629 vol.2, Dec. 1995. doi: 10.1109/CDC.1995.480371.

[30] WEN, J. T. Y., KREUTZ-DELGADO, K. "The attitude control problem", *IEEE Transactions on Automatic Control*, v. 36, n. 10, pp. 1148–1162, Oct. 1991. ISSN: 0018-9286. doi: 10.1109/9.90228.

[31] LIZARRALDE, F., WEN, J. T. "Attitude control without angular velocity measurement: a passivity approach", *IEEE Transactions on Automatic Control*, v. 41, n. 3, pp. 468–472, Mar. 1996. ISSN: 0018-9286. doi: 10.1109/9.486654.

[32] CABARBAYE, A., ESCOBAR, J. C., LOZANO, R., et al. "Fast adaptive control of a 3-DOF inertial stabilised platforms based on quaternions". In: *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 1463–1469, June 2017. doi: 10.1109/ICUAS.2017.7991426.

[33] MAO, J., YANG, J., LI, S., et al. "Output Feedback Stabilization of Inertial Stabilized Platform with Unmatched Disturbances Using Sliding Mode Approach", *20th IFAC World Congress*, v. 50, n. 1, pp. 5149 – 5154, 2017. ISSN: 2405-8963. doi: https://doi.org/10.1016/j.ifacol.2017.08.785. Disponível em: <http://www.sciencedirect.com/science/article/pii/S240589631731234X>.

[34] FENG, Y., YU, X., MAN, Z. "Non-singular terminal sliding mode control of rigid manipulators", *Automatica*, v. 38, n. 12, pp. 2159 – 2167, 2002. ISSN: 0005-1098. doi: 10.1016/S0005-1098(02)00147-4. Disponível em: <https://doi.org/10.1016/S0005-1098(02)00147-4>.

[35] LEITE, A. C. *Servovisão Adaptativa e Controle de Força para Robôs Manipuladores com Cinemática e Dinâmica Incertas Interagindo com Ambientes Não-Estruturados*. Tese de Doutorado, Instituto Alberto Luiz Coimbra de Pós-Graduação e Pesquisa de Engenharia, Aug. 2011. Disponível em: <http://www.pee.ufrj.br/index.php/pt/producao-academica/teses-de-doutorado/2011/2011082201-2011082201/file>.

[36] WANG, H. "Adaptive Control of Robot Manipulators With Uncertain Kinematics and Dynamics", *IEEE Transactions on Automatic Control*, v. 62, n. 2, pp. 948–954, Feb. 2017. ISSN: 0018-9286. doi: 10.1109/TAC.2016.2575827.

[37] SYMON, K. R. *Mechanics*. 3rd ed. , Addison-Wesley, 1971. ISBN: 0201073927.

[38] VILHENA ADORNO, B. "Robot Kinematic Modeling and Control Based on Dual Quaternion Algebra — Part I: Fundamentals." working paper or preprint, Feb. 2017. Disponível em: <https://hal.archives-ouvertes.fr/hal-01478225>.

[39] FROM, P. J. "An Explicit Formulation of Singularity-Free Dynamic Equations of Mechanical Systems in Lagrangian Form - Part one: Single Rigid Bodies", *Modeling, Identification and Control*, v. 33, n. 2, pp. 45–60, 2012. ISSN: 0332-7353. doi: 10.4173/mic.2012.2.2.

[40] FROM, P. J. "An Explicit Formulation of Singularity-Free Dynamic Equations of Mechanical Systems in Lagrangian Form - Part two: Multibody Systems", *Modeling, Identification and Control*, v. 33, n. 2, pp. 61–68, 2012. ISSN: 0332-7353. doi: 10.4173/mic.2012.2.3.

[41] YU, S., ZHAO, Y.-Z. "Simulation study on a friction compensation method for the inertial platform based on the disturbance observer", *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, v. 222, n. 3, pp. 341–346, 2008. doi: 10.1243/09544100JAERO291. Disponível em: <https://doi.org/10.1243/09544100JAERO291>.

[42] B. ARMSTRONG, C. D. W. "Friction Modeling and Compensation". In: *The Control Handbook*, CRC Press, Upper Saddle River, NJ, 1995.

[43] SHAH, S., SAHA, S., DUTT, J. K. "A new perspective towards decomposition of the generalized inertia matrix of multibody systems", *Multibody System Dynamics*, v. 43, n. 2, pp. 97–130, July 2017. ISSN: 1573-272X. doi: 10.

1007/s11044-017-9581-8. Disponível em: <https://doi.org/10.1007/s11044-017-9581-8>.

[44] GUO, Q., XU, Z. "Simulation of deep-water waves based on JONSWAP spectrum and realization by MATLAB". In: *2011 19th International Conference on Geoinformatics*, pp. 1–4, June 2011. doi: 10.1109/GeoInformatics.2011.5981100.

[45] UTKIN, V. I. "Sliding mode control design principles and applications to electric drives", *IEEE Transactions on Industrial Electronics*, v. 40, n. 1, pp. 23–36, Feb. 1993. ISSN: 0278-0046. doi: 10.1109/41.184818.

[46] CAMPA, R., CAMARILLO, K., ARIAS, L. "Kinematic Modeling and Control of Robot Manipulators via Unit Quaternions: Application to a Spherical Wrist". In: *Proceedings of the 45th IEEE Conference on Decision and Control*, pp. 6474–6479, Dec. 2006. doi: 10.1109/CDC.2006.377155.

[47] CHALANGA, A., KAMAL, S., FRIDMAN, L. M., et al. "Implementation of Super-Twisting Control: Super-Twisting and Higher Order Sliding-Mode Observer-Based Approaches", *IEEE Transactions on Industrial Electronics*, v. 63, n. 6, pp. 3677–3685, June 2016. ISSN: 0278-0046. doi: 10.1109/TIE.2016.2523913.

[48] MORENO, J. A., OSORIO, M. "Strict Lyapunov Functions for the Super-Twisting Algorithm", *IEEE Transactions on Automatic Control*, v. 57, n. 4, pp. 1035–1040, Apr. 2012. ISSN: 0018-9286. doi: 10.1109/TAC.2012.2186179.

[49] LEVANT, A. "Higher-order sliding modes, differentiation and output-feedback control", *International Journal of Control*, v. 76, n. 9-10, pp. 924–941, 2003. doi: 10.1080/0020717031000099029.

[50] HUANG, Y. J., KUO, T. C., CHANG, S. H. "Adaptive Sliding-Mode Control for NonlinearSystems With Uncertain Parameters", *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, v. 38, n. 2, pp. 534–539, Apr. 2008. ISSN: 1083-4419. doi: 10.1109/TSMCB.2007.910740.

[51] OLIVEIRA, T. R., MELO, G. T., HSU, L., et al. "Monitoring Functions Applied to Adaptive Sliding Mode Control for Disturbance Rejection", *20th IFAC World Congress*, v. 50, n. 1, pp. 2684 – 2689, 2017. ISSN: 2405-8963. doi: https://doi.org/10.1016/j.ifacol.2017.08.553.

Disponível em: <http://www.sciencedirect.com/science/article/pii/S2405896317309242>.

[52] F. REIS, M., CARVALHO, G., NEVES, A., et al. "Dynamic Model and Line of Sight Control of a 3-DOF Inertial Stabilization Platform via Feedback Linearization". In: *2018 IEEE American Control Conference*, pp. 1313–1318, June 2018.

[53] MORENO, J. A. "Lyapunov function for Levant's Second Order Differentiator". In: *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, pp. 6448–6453, Dec. 2012. doi: 10.1109/CDC.2012.6426877.

Appendix A

Quaternion Algebra

Let $R \in SO(3)$ be a *rotation matrix* describing the rotation from an arbitrary frame to another. Then, R is a *diffeomorphism* with respect to the projective space $\mathbb{RP}^3 = \left\{ \|v\|^2 \leq \pi \mid v \in \mathbb{R}^3 \right\}$. Therefore, each point $v \in \mathbb{RP}^3$ is a 4-parameter representation for $SO(3)$ called the *angle-axis*, where the unitary vector on the direction of v represents the rotation axis and $\|v\|$ represents the corresponding rotation angle around that axis.

Remark 22. *Note that $\mathbb{RP}^3$ covers $SO(3)$ twice, since any point on it actually represents the same rotation than the opposite point of the sphere.*

This representation can be expressed by $v = \{\theta, n\}$, where $\theta \in \mathbb{R}$ is the angle of rotation around the unit axis vector $n \in \mathbb{R}^3, \|n\| = 1$. Another non-minimal representation is the *unit quaternion*. The set of *quaternions* $\mathbb{H}$ is:

$$\begin{aligned} &\mathbb{H} := \{\eta + i\epsilon_1 + j\epsilon_2 + k\epsilon_3 \mid \eta, \epsilon_1, \epsilon_2, \epsilon_3 \in \mathbb{R}\} \,, \\ &i^2 = j^2 = k^2 = ijk = -1 \,. \end{aligned} \tag{A.1}$$

A quaternion $Q \in \mathbb{H}$ can also be represented as the pair

$$Q := \{\eta, \epsilon\} \,, \tag{A.2}$$

where $\eta = \mathsf{Re}(Q) \in \mathbb{R}$ represents the *real* part of the quaternion and $\epsilon = \mathsf{Im}(Q) = [\ \epsilon_1 \ \ \epsilon_2 \ \ \epsilon_3\ ]^\mathsf{T} \in \mathbb{R}^3$ represents the vector part. The quaternion *conjugate* is given by

$$Q^* = \{\eta, -\epsilon\} \,. \tag{A.3}$$

One can also represent the quaternion in fully vector form by the notation $\bar{Q} = [\ \eta \ \ \epsilon_1 \ \ \epsilon_2 \ \ \epsilon_3\ ]^\mathsf{T} \in \mathbb{R}^4$.

Quaternions also form an algebraic *group* with respect to *multiplication*. Given

two quaternions $Q_1 = \{\eta_1, \epsilon_1\}$ and $Q_2 = \{\eta_2, \epsilon_2\}$, their multiplication follows the rules established by (A.1):

$$Q_1 \circ Q_2 = \{\eta_1\eta_2 - \epsilon_1^{\mathsf{T}}\epsilon_2,\, \eta_1\epsilon_2 + \eta_2\epsilon_1 + \epsilon_1 \times \epsilon_2\}\,. \tag{A.4}$$

Quaternion multiplication can also be performed as the linear transformation in $\mathbb{R}^4$

$$\overline{Q_1 \circ Q_2} = \mathbf{H}_+(Q_1)\,\bar{Q}_2\,, \tag{A.5}$$

$$= \mathbf{H}_-(Q_2)\,\bar{Q}_1\,, \tag{A.6}$$

where $\mathbf{H}_+$, $\mathbf{H}_-$ are *Hamilton operators* defined by

$$\mathbf{H}_\pm(Q) = \left[\overline{Q} \;\; \mathbf{h}_\pm(Q)\right],\; \mathbf{h}_\pm(Q) = \begin{bmatrix} -\epsilon^{\mathsf{T}} \\ \eta\,\mathbf{I}_3 \pm \widehat{\epsilon} \end{bmatrix}. \tag{A.7}$$

The square of the quaternion *norm* is defined as the *scalar*

$$\|Q\|^2 = Q \circ Q^* = \{\eta^2 + \epsilon^{\mathsf{T}}\epsilon, 0\}\,, \tag{A.8}$$

and its *inverse* is the quaternion Q^{-1} such that $Q \circ Q^{-1} = \mathsf{I}_Q = \{1, 0\}$, the *unitary* quaternion. The set of *unit quaternions* $\mathbb{H}^* = \{Q \in \mathbb{R} : \|Q\| = 1\}$ can be used as a parametrization for orientation in the following way. For an element $p = \{\theta, n\} \in \mathbb{RP}$, define

$$Q = \left\{cos\left(\frac{\theta}{2}\right)\,, sin\left(\frac{\theta}{2}\right)\,n\right\} \in \mathbb{H}^*\,, \tag{A.9}$$

which clearly has unit norm.

Remark 23. *The inverse of an unit quaternion is given by $Q^{-1} = Q^*$, which according to* (A.9)*, corresponds to the opposite rotation due to negative direction of the rotation axis n.*

Let $r_0, r_1, ..., r_n \in \mathbb{H}^*$ be the n absolute rotations between frames $\mathbf{E}_0, \mathbf{E}_1, ..., \mathbf{E}_n$ and the world frame $\mathbf{E}_w$, and $r_{i+1}^i \in \mathbb{H}^*$ $(i = 1, 2, ..., n-1)$ represent the rotations from frame $\mathbf{E}_i$ to $\mathbf{E}_{i+1}$. Since the unit quaternions form a group with respect to multiplication, then

$$r_n = r_1 \circ r_2^1 \circ ... \circ r_n^{n-1} \in \mathbb{H}^*\,. \tag{A.10}$$

Now, define the set of *pure* quaternions $\mathbb{H}_p = \{v \in \mathbb{H} : \mathsf{Re}(v) = 0\}$. Note that any vector from $\mathbb{R}_3$ can be represented as the vector part of a corresponding element $v \in \mathbb{H}_p$. Let v^i and $v^j \in \mathbb{H}_p$ be representations for a vector $\vec{v}$ in frames $\mathbf{E}_i$ and $\mathbf{E}_j$,

respectively, and r_j^i represents the rotation from $\mathbf{E}_i$ to $\mathbf{E}_j$, with unitary axis $n_j^i \in \mathbb{R}^3$ and rotation angle θ_{ij}. Then, the following relation holds:

$$v^i = (r_j^i) \circ v^j \circ (r_j^i)^* = Ad_{r_j^i}\left[v^j\right] , \tag{A.11}$$

where $Ad_{r_j^i}[*]$ is the adjoint *operator*. Note that, in vector algebra, $Ad_{r_j^i}$ represents the corresponding rotation matrix $R_{ij} \in SO(3)$ associated to the unit quaternion $r_j^i \in \mathbb{H}^*$. In terms of the axis and angle of r_j^i, this matrix is given by

$$R_{ij} = N_j^i + s_{ij}\,\mathbf{S}(n_j^i) + c_{ij}\,(\mathsf{I}_3 - N_j^i) \tag{A.12}$$

where $N_j^i = n_j^i (n_j^i)^\mathsf{T}$ and s_{ij} and c_{ij} are the sine and cosine functions of θ_{ij}. The rotation matrix corresponding to an absolute rotation $r_i \in \mathbb{H}^*$ (with respect to the world frame) is written with only one subscript, as $R_i \in SO(3)$. In terms of the components of $r_i = \{\eta_i, \epsilon_{i_x}, \epsilon_{i_y}, \epsilon_{i_z}\}$, this matrix is given by [20]:

$$R_i = \begin{bmatrix} 2(\eta_i^2 + \epsilon_{i_x}^2) - 1 & 2(\epsilon_{i_x}\epsilon_{i_y} - \eta_i\epsilon_{i_z}) & 2(\epsilon_{i_x}\epsilon_{i_z} + \eta_i\epsilon_{i_y}) \\ 2(\epsilon_{i_x}\epsilon_{i_y} + \eta_i\epsilon_{i_z}) & 2(\eta_i^2 + \epsilon_{i_y}^2) - 1 & 2(\epsilon_{i_y}\epsilon_{i_z} - \eta_i\epsilon_{i_x}) \\ 2(\epsilon_{i_x}\epsilon_{i_z} - \eta_i\epsilon_{i_y}) & 2(\epsilon_{i_y}\epsilon_{i_z} + \eta_i\epsilon_{i_x}) & 2(\eta_i^2 + \epsilon_{i_z}^2) - 1 \end{bmatrix} . \tag{A.13}$$

The opposite mapping can be found as [20]:

$$\begin{aligned} \eta_i &= \frac{1}{2}\sqrt{r_{11} + r_{22} + r_{33} + 1}\,, \\ \epsilon_i &= \frac{1}{2}\begin{bmatrix} sgn(r_{32} - r_{23})\sqrt{r_{11} - r_{22} - r_{33} + 1} \\ sgn(r_{13} - r_{31})\sqrt{r_{22} - r_{33} - r_{11} + 1} \\ sgn(r_{21} - r_{12})\sqrt{r_{33} - r_{11} - r_{22} + 1} \end{bmatrix} , \end{aligned} \tag{A.14}$$

where $R_i = [\, r_{kj} \,], k, j = 1, 2, 3$.

Now, let $\vec{v}_i$ and $\vec{\omega}_i$ be the physical linear and angular velocities of $\mathbf{E}_i$. They are represented by $v_i \in \mathbb{R}^3$, $\omega_i \in \mathbb{R}^3$ when written in the world frame and by $v_i^i \in \mathbb{R}^3$, $\omega_i^i \in \mathbb{R}^3$ when written in its own body frame. Let $r_i = \{\eta_i\,, \epsilon_i\} \in \mathbb{H}^*$ be the absolute rotation of $\mathbf{E}_i$. The time-derivative of r_i can be related to ω_i^i by

$$\dot{r}_i = \frac{1}{2}\,\omega_i \circ r_i\,, \tag{A.15}$$

$$= \frac{1}{2}\,r_i \circ \omega_i^i\,, \tag{A.16}$$

which is known as the *quaternion propagation* formula [38], and can also be expressed

in vector form by

$$\dot{\bar{r}}_i = \begin{bmatrix} \dot{\eta}_i \\ \dot{\epsilon}_i \end{bmatrix} = \frac{1}{2}\,\mathbf{h}_-(r_i)\,\omega_i\,, \tag{A.17}$$

$$= \frac{1}{2}\,\mathbf{h}_+(r_i)\,\omega_i^i\,. \tag{A.18}$$

Appendix B

Proofs of Theorems

B.1 Proof of Lemma 1

Proof. Using (2.87) and (2.83), the Lagragian as a function of $x, \dot{x} \in \mathbb{R}^N$ is given by

$$\mathcal{L}(x, \dot{x}) = \frac{1}{2}\, \dot{x}^{\mathsf{T}}\, S^{\mathsf{T}}(x)\, M(x)\, S(x)\, \dot{x} - \mathcal{U}(x)\,. \tag{B.1}$$

By (2.82) and (2.85), the Lagrange equations can be expressed by

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}(x, \dot{x})}{\partial \dot{x}}\right) - \frac{\partial \mathcal{L}(x, \dot{x})}{\partial x} = S^{\mathsf{T}}(x)\, \tau\,, \tag{B.2}$$

Computing the terms of (B.2) from (B.1), yields

$$\frac{\partial \mathcal{L}}{\partial \dot{x}} = S^{\mathsf{T}} M\, S\, \dot{x}\,, \tag{B.3}$$

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{x}}\right) = (\dot{S}^{\mathsf{T}} M\, S + S^{\mathsf{T}} \dot{M}\, S + S^{\mathsf{T}} M\, \dot{S})\, \dot{x} + S^{\mathsf{T}} M\, S\, \ddot{x}\,, \tag{B.4}$$

$$\frac{\partial \mathcal{L}}{\partial x} = \frac{1}{2}\, \frac{\partial^{\mathsf{T}} v}{\partial x}\, M\, v + \frac{1}{2}\, \frac{\partial^{\mathsf{T}} (M\, v)}{\partial x}\, v - \frac{\partial \mathcal{U}}{\partial x}\,. \tag{B.5}$$

Substituting (B.3), (B.4) and (B.5) into (B.2) and left-multiplying the resulting equation by $S^{-\mathsf{T}}(x) \in \mathbb{R}^{N \times N}$ results in (2.88).

□

B.2 Proof of Theorem 1

Proof. Comparing (2.89) to (2.88), the terms $M(x)\,\dot{v}$ and $G(\xi) = S^{-\mathsf{T}}(x)\,\frac{\partial \mathcal{U}(x)}{\partial x}$ are easily identified. However, the Coriolis term is given by

$$C(x,v) = \dot{M}(x)\,v - \frac{1}{2}\,S^{-\mathsf{T}}(x)\,\frac{\partial^{\mathsf{T}}(M(x)\,v)}{\partial x}\,v + S^{-\mathsf{T}}(x)\,\left(\dot{S}^{\mathsf{T}}(x) - \frac{1}{2}\frac{\partial^{\mathsf{T}} v}{\partial x}\right)\,M(x)\,v\,. \tag{B.6}$$

The expression for $C(x,v)$ in (2.90) can be found by performing an element-wise expansion on each term of (B.6), using (2.83):

$$\dot{M}_{ij}(x) = \sum_{l}\frac{\partial M_{ij}}{\partial x_l}\,\dot{x}_l = \sum_{l,k}\frac{\partial M_{ij}}{\partial x_l}\,S^{-1}_{lk}\,v_k\,, \tag{B.7}$$

$$\begin{aligned}\left(S^{-\mathsf{T}}(x)\,\frac{\partial^{\mathsf{T}}(M(x)\,v)}{\partial x}\right)_{ij} &= \sum_{l} S^{-1}_{li}\,\frac{\partial}{\partial x_l}\left(\sum_{k} M_{jk}\,v_k\right)\\ &= \sum_{l,k} S^{-1}_{li}\left(\frac{\partial M_{jk}}{\partial x_l}\,v_k + M_{jk}\,\frac{\partial v_k}{\partial x_l}\right)\\ &= \sum_{l,k} S^{-1}_{li}\,\frac{\partial M_{jk}}{\partial x_l}\,v_k\\ &\quad + \sum_{k}\left(\sum_{l,m,s} S^{-1}_{li}\,\frac{\partial S_{sm}}{\partial x_l}\,S^{-1}_{mk}\right)M_{js}\,v_k\,,\end{aligned} \tag{B.8}$$

$$\left(S^{-\mathsf{T}}\,\dot{S}^{\mathsf{T}}\,M(x)\right)_{is} = \sum_{k}\left(\sum_{l,m,s} S^{-1}_{li}\,\frac{\partial S_{sl}}{\partial x_m}\,S^{-1}_{mk}\right)M_{sj}\,v_k\,, \tag{B.9}$$

$$\left(S^{-\mathsf{T}}\,\frac{\partial^{\mathsf{T}} v}{\partial x}\,M(x)\right)_{is} = \sum_{k}\left(\sum_{l,m,s} S^{-1}_{li}\,\frac{\partial S_{sm}}{\partial x_l}\,S^{-1}_{mk}\right)M_{sj}\,v_k\,, \tag{B.10}$$

where S^{-1}_{ij} are the elements of the inverse mapping $S^{-1}(x)$. Next, summing up (B.7), (B.8), (B.9) and (B.10) according to (B.6) and grouping the matrices before v_k, the expressions on (2.91) and (2.92) emerge. This demonstrates (2.90).

□

B.3 Proof of Theorem 3

Proof. The errors of Fig. 3.2 are defined as

$$e_1 = u_1 - y_2, \quad e_2 = u_2 + y_1\,. \tag{B.11}$$

Then, using (B.11) and Assumptions (i), (ii)

$$\begin{aligned}\|e_1\| &\leq \|u_1\| + A\,\|e_2\|^2 + B\,\|e_2\| + C \\ &\leq \|u_1\| + A(\|u_2\| + \gamma\,\|e_1\| + \beta)^2 + B(\|u_2\| + \gamma\,\|e_1\| + \beta) + C \\ &\leq \gamma^2 A\,\|e_1\|^2 + (\gamma B + 2\gamma A\beta + 2\gamma A\,\|u_2\|)\,\|e_1\| + D_1\,.\end{aligned} \tag{B.12}$$

The inequality (B.12) can be rewritten as

$$\gamma^2 A\,\|e_1\|^2 + \underbrace{(\gamma B + 2\gamma A\beta + 2\gamma A\,\|u_2\| - 1)}_{b_1}\,\|e_1\| + D_1 \geq 0\,. \tag{B.13}$$

It describes a quadratic inequality in $\|e_1\|$, illustrated in Fig. B.1. Numbers ϵ_1 and

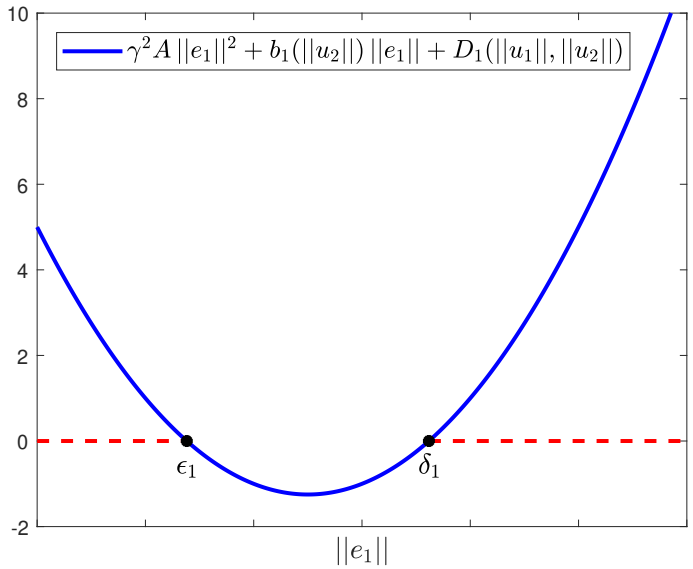


Figure B.1: Quadratic function of the error norm.

δ_1 are the roots of the quadratic function in the left-hand side of (B.13). Note that the two dashed horizontal lines represent the subsets of $\mathbb{R}_{>0}$ $S_1 = \{\|e_1\| \leq \epsilon_1\}$ and $S_2 = \{\|e_1\| \geq \delta_1\}$ in which (B.13) is satisfied. Therefore, under conditions 3.9 and 3.10, $\|e_1\| \in S_1 \cup S_2$. Note that $S_1 \cup S_2 = \mathbb{R}_{>0}$ if $\epsilon_1 = \delta_1$ or if the quadratic function has no real roots. Then, if

1. $b_1 < 0\,,$
2. $4\gamma^2 A\,D_1 < b_1^2\,,$

the roots ϵ_1 and δ_1 are both real and positive. These are the first two Assumptions (i) and (ii) of the theorem. Then, by (B.13), if $\|e_1(0)\| \in S_1$ at time $t = 0$, then $\|e_1\| \in S_1 \;\; \forall t > 0$. Graphically, it means that $\|e_1\|$ stays in the left dashed subset of $\mathbb{R}_{>0}$ in Fig. B.1.

Using similar arguments, it can be shown that, under Assumptions (i) and (ii), the inequality

$$\gamma A \, \|e_2\|^2 + \underbrace{(\gamma B - 1)}_{b_2} \|e_2\| + D_2 \geq 0 \tag{B.14}$$

is valid. Again, if

1. $b_2 < 0$,
2. $4\gamma A \, D_2 < b_2^2$,

the roots ϵ_2 and δ_2 of the quadratic function on the left-hand side of (B.14) are both real and positive, where $\epsilon_2 < \delta_2$ by definition. The first condition $b_2 = \gamma B - 1 < 0$ is a weaker version of previous condition $b_1 < 0$, and if this one is satisfied, the former will also be. However, the second condition is the third Assumption (iii) of the theorem. Then, as before, by (B.14), if $\|e_2(0)\| \leq \epsilon_2$ at time $t = 0$, then $\|e_2\| \leq \epsilon_2 \;\; \forall t > 0$.

Since u_1 and u_2 are bounded, then the boundedness of e_1 and e_2 imply in the boundedness of the outputs y_1 and y_2. Finally, from the triangle inequality, $\|y\| \leq \|y_1\| + \|y_2\|$, which completes the proof.

□

B.4 Proof of Theorem 4

Proof. First, note that Assumptions (i) and (ii) are needed so that the terms $\widehat{M}_\eta$ and $\widehat{\tau}_d^c$ can be computed in (3.15). Then, from (3.14), (3.15) and the plant dynamics (3.2), the RPY orientation dynamics can be written as

$$\ddot{\eta}_{c_2} = G(\eta_{c_2}, t) \underbrace{\left(\ddot{\eta}_{d_2} + K_D \, \dot{e}_\eta + K_P \, e_\eta + K_I \int_0^t e_\eta(\tau) \, d\tau \right)}_{\text{PID controller + feedforward term}} - d_\eta(\eta_{c_2}, \dot{\eta}_{c_2}, t) \, . \tag{B.15}$$

where $G(\eta_{c_2}, t) = (M_\eta + \Delta M_{qq}^c J_q^{-1})^{-1} \, \widehat{M}_\eta$ is a state and time dependent gain matrix that is close to the identity matrix if the parametric error is small, and d_η is a state and time dependent disturbance.

Remark 24. *Note that* (B.15) *is a double integrator with a PID controller and a feedforward RPY acceleration term multiplied by a high-frequency gain* $G(\eta_{c_2}, t) \approx \mathsf{I}_3$ *with an input disturbance due to the parametric error.*

Therefore, (B.15) can be written as a feedback interconnection of two BIBO stable subsystems H_1 and H_2, as shown in Fig. 3.2.

Considering the approximation $G(\eta_{c_2}, t) \approx \mathsf{I}_3$, H_1 is the stable, unperturbed system

$$\eta_{c_2} \approx \mathcal{L}^{-1}\{S(s)\, D_\eta(s)\}\,,$$

where $\mathcal{L}^{-1}$ stands for the inverse Laplace transform operator, $D_\eta(s)$ is the Laplace transform of the disturbance d_η and $S(s)$ is the *sensitivity* transfer function of the system. It gives the relation in the frequency domain between the input disturbance d_η and the output η_{c_2}. For (B.15), it is given by

$$S_i(s) = \frac{s}{s^3 + K_{D_i} s^2 + K_{P_i} s + K_{I_i}}\,,$$

where K_{P_i}, K_{D_i} and K_{I_i} $(i = 1, 2, 3)$ are the elements on the diagonals of the gain matrices K_P, K_D and K_I, respectively. Each transfer function $S_i(s)$ represents the sensitivity of each RPY channel with respect to input disturbances. By [8],

$$\|\eta_{c_2}\| \leq \gamma_1 \|d_\eta\| + \beta_1\,, \quad \gamma_1 = \sup_{\omega \in \mathbb{R}} \|S(j\omega)\|\,, \tag{B.16}$$

$$\|\dot{\eta}_{c_2}\| \leq \gamma_2 \|d_\eta\| + \beta_2\,, \quad \gamma_2 = \sup_{\omega \in \mathbb{R}} \|s\, S(j\omega)\|\,. \tag{B.17}$$

Remark 25. *The stability of system* (B.15) *with* $d_\eta = 0$ *and* $G(\eta_{c_2}, t) \neq \mathsf{I}_3$ *can be demonstrated using the circle criteria [8].*

Therefore, defining the camera orientation state as $\eta = \begin{bmatrix} \eta_{c_2} \\ \dot{\eta}_{c_2} \end{bmatrix}$, it can be seen from (B.16) and (B.17) that

$$\begin{aligned} \|\eta\| &\leq \|\eta_{c_2}\| + \|\dot{\eta}_{c_2}\| \\ &\leq \underbrace{(\gamma_1 + \gamma_2)}_{\gamma} \|d_\eta\| + \underbrace{(\beta_1 + \beta_2)}_{\beta}\,. \end{aligned} \tag{B.18}$$

The subsystem H_2 is the state-dependent disturbance $d_\eta(\eta, t)$. Since it is composed of Coriolis, centrifugal and gravity terms, it depends *quadratically* on $\dot{\eta}_{c_2}$. Therefore, using Assumptions (i) and (iii) and the linearity of the disturbance d_η with respect to the dynamic parameters, it is possible to find positive constants A, C such that

$$\|d_\eta\| \leq \left(A \|\eta\|^2 + C\right) \left\|\widetilde{\Pi}_g\right\|\,, \tag{B.19}$$

where constant C is strongly dependent on the maximum vehicle velocities and

accelerations.

Then, (B.15) is a feedback connection between the two BIBO stable systems H_1 and H_2, as illustrated by Fig. 3.2 with $u_1 = u_2 = 0$, under conditions (B.18) and (B.19).

Finally, using Theorem 3 with $\|u_1\| = \|u_2\| = 0$ and term $B = 0$, the *sufficient conditions* for the existence of an attractive domain for (B.15) are equivalent to

$$\left\|\widetilde{\Pi}_g\right\| < \frac{A\beta}{2\gamma}\,, \tag{B.20}$$

$$\left\|\widetilde{\Pi}_g\right\| < \frac{\sqrt{a^2\beta^2 + AC} - A\beta}{2\gamma AC}\,. \tag{B.21}$$

Clearly, the maximum admissible error for the dynamic parameters is strongly dependent on the bounds for the vehicle velocities, accelerations and RPY reference.

Then, under conditions (B.20), (B.21), due to (B.18) and (B.19), the norm of the RPY error e_η will be bounded by a set of the order $\mathcal{O}(\left\|\widetilde{\Pi}_d\right\|)$.

□

B.5 Proof of Theorem 5

Proof. First, note that Assumption (i) is needed so that $\widehat{J}_q^{\,-1}$ can be computed in (3.21). Substituting the control laws (3.20), (3.21) and (3.19) into the RPY dynamics gives:

$$\ddot{\eta}_{c_2} = G(\eta_{c_2}, t)\,(\ddot{\eta}_{d_2} + PID) - d_\eta(\eta_{c_2}, \dot{\eta}_{c_2}, t)\,, \tag{B.22}$$

where $G(\eta_{c_2}, t) = J_q\, {M^0_{qq}}^{-1}\, \widehat{M}^0_{qq}\, \widehat{J}_q^{-1}$ is a state and time dependent gain matrix that is close to the identity matrix if the parametric error is small, and d_η is a state and time dependent disturbance. Therefore, by the same arguments from Theorem 4, (B.22) can once again be written as a feedback interconnection of two BIBO stable subsystems H_1 and H_2, as shown in Fig. 3.2, where H_1 is the stable, unperturbed system

$$\eta_{c_2} \approx \mathcal{L}^{-1}\{S(s)\, D_\eta(s)\}\,,$$

and H_2 is the disturbance, which depends quadratically on $\dot{\eta}_{c_2}$.

As before, due to Assumption (ii) and the linearity of the disturbance d_η with

respect to the *geometric and dynamic parameters*, we have

$$\|\eta\| \leq \gamma \|d_\eta\| + \beta \,, \tag{B.23}$$

$$\|d_\eta\| \leq \left(A \|\eta\|^2 + C\right) \left\|\widetilde{\Pi}\right\| \,. \tag{B.24}$$

Again, using Theorem 3 with $\|u_1\| = \|u_2\| = 0$ and term $B = 0$, the *sufficient conditions* for the existence of an attractive domain for (B.22) are equivalent to

$$\left\|\widetilde{\Pi}\right\| < \frac{A\beta}{2\gamma} \,, \tag{B.25}$$

$$\left\|\widetilde{\Pi}\right\| < \frac{\sqrt{a^2\beta^2 + AC} - A\beta}{2\gamma AC} \,. \tag{B.26}$$

Then, under conditions (B.25), (B.26), due to (B.23) and (B.24), the norm of the RPY error e_η will be bounded by a set of the order $\mathcal{O}(\left\|\widetilde{\Pi}\right\|)$.

□

B.6 Proof of Theorem 6

Proof. First, define the following Lyapunov function candidate:

$$V(e_c) = (\eta_d - \eta_c)^2 + (\epsilon_d - \epsilon_c)^\mathsf{T}(\epsilon_d - \epsilon_c) \geq 0 \,, \tag{B.27}$$

where the quaternion error was defined in (4.19). The time derivative of (B.27) is

$$\dot{V} = 2\,(\eta_d - \eta_c)\,(\dot{\eta}_d - \dot{\eta}_c) + 2\,(\epsilon_d - \epsilon_c)\,(\dot{\epsilon}_d - \dot{\epsilon}_c) \,.$$

Using the quaternion propagation formula (A.18) and performing additional simplifications, yields:

$$\dot{V} = (\eta_c\,\epsilon_d - \eta_d\,\epsilon_c + \epsilon_d \times \epsilon_c)\,e_\omega = \mathsf{Im}(e_c)^\mathsf{T}\,e_\omega \,.$$

Finally, applying the first-order dynamics of (4.21):

$$\dot{V} = -\mathsf{Im}(e_c)^\mathsf{T}\,K_c\,\mathsf{Im}(e_c) \leq 0 \,. \tag{B.28}$$

Since (B.28) is only negative semi-definite, LaSalle's invariance principle can be used to conclude about the asymptotic stability of $e_c = \mathsf{I}_Q$. The invariant set D is defined

as the *null space* of $\dot{V}$, which by (B.28) is given by

$$D = \{e_c \in \mathbb{H}^* \mid \mathsf{Im}(e_c) = 0\}\,.$$

It contains two points, $e_c = \pm \mathsf{I}_Q$. However, only $e_c = \mathsf{I}_Q$ is a stable solution for (4.21). Therefore, we conclude that the quaternion error $e_c \in \mathbb{H}^*$ must tend asymptotically to I_Q, which by (4.19) means that $r_c \to r_{c_d}$ asymptotically. Furthermore, since the Lyapunov function is *radially unbounded*, we conclude that $e_c = \mathsf{I}_Q$ is actually *globally* asymptotically stable.

□

B.7 Proof of Theorem 7

Proof. Using (4.15) and Assumption 18, the dynamics of the sliding variable s_x is given by

$$\begin{aligned}\dot{s}_x &= \dot{x}_2 - w(t) \\ &= {M_{qq}^0}^{-1} u(t) + x_3 - w(t)\,. \end{aligned} \tag{B.29}$$

Substituting (4.26) into (B.29), it becomes

$$\dot{s}_x = (\mathsf{I}_3 - {M_{qq}^0}^{-1} \Delta M_{qq}) \left(w(t) - \Lambda_1 \lfloor s_x \rceil^{1/2} - \Lambda_2 \int_0^t \lfloor s_x \rceil^0 \, d\tau \right) + x_3 - w(t) \tag{B.30}$$

$$= -\Lambda_1 \lfloor s_x \rceil^{1/2} - \Lambda_2 \int_0^t \lfloor s_x \rceil^0 \, d\tau + x_3 - {M_{qq}^0}^{-1} \Delta M_{qq} \, \widehat{M_{qq}^0}^{-1} u + x_3 \tag{B.31}$$

where $\Delta M_{qq}^0 = M_{qq}^0 - \widehat{M}_{qq}^0$. Then, it is possible to rewrite (B.31) as

$$\begin{aligned} \dot{s}_x &= -\Lambda_1 \lfloor s_x \rceil^{1/2} + w_x\,, \\ \dot{w}_x &= -\Lambda_2 \lfloor s_x \rceil^0 + d_x\,, \\ d_x &= \frac{d}{dt}\left({M_{qq}^0}^{-1} \Delta M_{qq} \, \widehat{M_{qq}^0}^{-1} \right) u + {M_{qq}^0}^{-1} \Delta M_{qq}^0 \, \widehat{M_{qq}^0}^{-1} \dot{u} + \dot{x}_3\,, \end{aligned} \tag{B.32}$$

where the disturbance d_x is clearly dependent on the base motion, the states, control signal and on the errors on the parametric errors. Note also that the dependance on control is conditioned to the existence of uncertainty in the computation of the mass matrices, due to the non-vanishing term ΔM_{qq}^0.

Note that (B.33) is in the form of the super-twisting algorithm (STA), which is finite-time stable for bounded matched disturbances. It is evident that, if the nominal parameters are known, system (B.31) is only perturbed by $d_x \approx \dot{x}_3$. Due to

Assumptions (ii), (iii) and the form of control law (4.26), the following inequalities hold:

$$\left\| \frac{d}{dt} \left({M_{qq}^0}^{-1} \, \Delta M_{qq} \, \widehat{M_{qq}^0}^{-1} \right) u \right\| < L_{x_1} \,, \tag{B.33}$$

$$\left\| {M_{qq}^0}^{-1} \, \Delta M_{qq}^0 \, \widehat{M_{qq}^0}^{-1} \, \dot{u} \right\| < L_{x_2} \,, \tag{B.34}$$

$$\|\dot{x}_3\| < L_{x_3} \,. \tag{B.35}$$

Then, $\|d_x\| < L_{x_1} + L_{x_2} + L_{x_3}$, and according to [48], it is possible to chose Λ_1 and Λ_2 so that (B.33) achieves SOSM in finite-time. It means that after a time $T_1 > 0$, $s_x = \dot{s}_x = 0$ and due to (B.29), $\dot{x}_2 = w(t)$ $\forall t > T_1$.

Next, using (4.16), (4.19) and (4.20), the dynamics of the tracking sliding variable (4.23) is given by

$$\dot{s}_y = \dot{y}_{2_d} - J_{0c_2}^c(x_1) \, \dot{x}_2 - y_3 + K_c \, \psi(y_1, y_2, r_{c_d}) \,, \tag{B.36}$$

with $\dot{y}_{1_d} = \mathbf{h}_-(y_1) \, y_{2_d}$. Since $\dot{x}_2 = \dot{s}_x + w(t)$, substituting (4.24) into (B.36) and using Assumption (i) yields

$$\begin{aligned} \dot{s}_y &= -\Lambda_3 \, \lfloor s_y \rceil^{1/2} + w_y \,, \\ \dot{w}_y &= -\Lambda_4 \, \lfloor s_y \rceil^0 - d_y \,, \end{aligned} \tag{B.37}$$

$$d_y = \dot{y}_3 + \frac{d}{dt} \left(J_{0c_2}^c \, \dot{s}_x \right) + \frac{d}{dt} \left(W_\omega^* \right) \widetilde{\Pi}_g \,, \tag{B.38}$$

with $W_\omega^* = W_\omega(x_1, w(t), 0)$, according to (2.57). Again, due to Assumptions (ii), (iii) and the form of control law (4.24), we have

$$\left\| \frac{d}{dt} \left(J_{0c_2}^c \, \dot{s}_x \right) \right\| < L_{y_1} \,, \tag{B.39}$$

$$\left\| \frac{d}{dt} \left(W_\omega^* \right) \widetilde{\Pi}_g \right\| < L_{y_2} \,, \tag{B.40}$$

$$\|\dot{y}_3\| < L_{y_3} \,. \tag{B.41}$$

Note that (B.39) is true because $\ddot{s}_x$ is bounded, but constant L_{y_1} clearly depends on the initial conditions of (4.15). Also, in (B.40), the time derivative of W_ω^* depends on x_1, x_2, $w(t)$ and $\dot{w}(t)$, which are also bounded. Then, $\|d_y\| < L_{y_1} + L_{y_2} + L_{y_3}$, again guaranteeing finite-time stabilization of (B.38) after a time $T_2 > 0$. It means that for all $t \geq T_2$, the tracking system is sliding and therefore, it follows the nonlinear dynamics of the sliding variable (4.23), which is asymptotically stable [20]. Therefore, the quaternion errors (4.19) and (4.20) tend to I_Q and zero (respectively) asymptotically after a time $max(T_1, T_2)$.

□

B.8 Proof of Theorem 8

Proof. Using (4.15) and (4.28), the dynamics of the estimation errors is

$$\begin{aligned}
\dot{e}_{x_1} &= -K_1 \lfloor e_{x_1} \rceil^{2/3} + e_{x_2} \,, \\
\dot{e}_{x_2} &= -K_2 \lfloor e_{x_1} \rceil^{1/3} + e_{x_3} + (M_{qq}^{0\,-1} - \widehat{M_{qq}^{0}}^{-1})\, u \,, \\
\dot{e}_{x_3} &= -K_3 \lfloor e_{x_1} \rceil^{0} + \dot{x}_3 \,.
\end{aligned} \tag{B.42}$$

By defining $e_{x_4} = e_{x_3} + \left(M_{qq}^{0\,-1} - \widehat{M_{qq}^{0}}^{-1}\right) u$, it is possible to rewrite (B.42) as

$$\begin{aligned}
\dot{e}_{x_1} &= -K_1 \lfloor e_{x_1} \rceil^{2/3} + e_{x_2} \,, \\
\dot{e}_{x_2} &= -K_2 \lfloor e_{x_1} \rceil^{1/3} + e_{x_4} \,, \\
\dot{e}_{x_4} &= -K_3 \lfloor e_{x_1} \rceil^{0} + d_e \,, \\
d_e &= \dot{x}_3 + \left(M_{qq}^{0\,-1} - \widehat{M_{qq}^{0}}^{-1}\right) \dot{u} + \frac{d}{dt}\left(M_{qq}^{0\,-1} - \widehat{M_{qq}^{0}}^{-1}\right) u \,.
\end{aligned} \tag{B.43}$$

Due to Assumption (ii) and (4.30), two constants $L_{e_1}, L_{e_2} > 0$ exist, such that

$$\left\| \left(M_{qq}^{0\,-1} - \widehat{M_{qq}^{0}}^{-1}\right) \dot{u} \right\| < L_{e_1} \,, \tag{B.44}$$

$$\left\| \frac{d}{dt}\left(M_{qq}^{0\,-1} - \widehat{M_{qq}^{0}}^{-1}\right) u \right\| < L_{e_2} \,. \tag{B.45}$$

Also, by Assumption (iii), $\|\dot{x}_3\| < L_{x_3}$ also holds, in (B.35). Then, $\|d_e\| < L_{e_1} + L_{e_2} + L_{x_3}$, and therefore the disturbance d_e is uniformly norm-bounded. According to [53], it is possible to chose K_1, K_2 and K_3 so that the states on (B.44) are finite-time stable.

Remark 26. *Since* $M_{qq}^{0\,-1} - \widehat{M_{qq}^{0}}^{-1} \neq 0$ *due to parametric uncertainty, the estimation error* e_{x_3} *is expected to be norm-bounded only. Therefore,* $x_3 = \widehat{x}_3 + \beta(\widetilde{\Pi})$*, where* $\beta(\widetilde{\Pi}) = \left(M_{qq}^{0\,-1} - \widehat{M_{qq}^{0}}^{-1}\right) u$ *is a residue dependent on the parametric uncertainty.*

The dynamics of the modified sliding variable is given by

$$\dot{\widehat{s}}_x = K_2 \lfloor e_{x_1} \rceil^{1/3} + \widehat{x}_3 + \widehat{M_{qq}^{0}}^{-1} u(t) - w(t) \,. \tag{B.46}$$

Using the *continuous* control law (4.30), yields

$$\begin{aligned}\dot{\widehat{s}}_x &= -\Lambda_1 \lfloor \widehat{s}_x \rceil^{1/2} + \widehat{w}_x \,, \\ \dot{\widehat{w}}_x &= -\Lambda_2 \lfloor \widehat{s}_x \rceil^{0} + K_3 \lfloor e_{x_1} \rceil^{0} \,. \end{aligned} \tag{B.47}$$

Since the disturbance $K_3 \lfloor e_{x_1} \rceil^0$ is obviously norm-bounded, the STA (B.47) is finite-time stable. Therefore, after a finite time $\bar{T}_1 > 0$, $\dot{x}_2 = w(t)$.

To prove the stability of the tracking controller, a similar procedure is performed. Since $\dot{x}_2 = \dot{\widehat{s}}_x + \dot{e}_{x_2} + w(t)$, substituting (4.24) into (B.36) and using Assumption (i), yields

$$\begin{aligned}\dot{s}_y &= -\Lambda_3 \lfloor s_y \rceil^{1/2} + w_y \,, \\ \dot{w}_y &= -\Lambda_4 \lfloor s_y \rceil^{0} - \bar{d}_y \,, \\ \bar{d}_y &= \dot{y}_3 + \frac{d}{dt}\left(J^c_{0c_2} \, \dot{\widehat{s}}_x \right) - \frac{d}{dt}\left(J^c_{0c_2} \dot{e}_{x_2} \right) + \frac{d}{dt}\left(W^*_\omega \right) \widetilde{\Pi}_g \,. \end{aligned} \tag{B.48}$$

Again, due to Assumptions (i) and (ii), (B.44) and (B.47), two positive constants $\bar{L}_{y_1}$, $\bar{L}_{y_2}$ exist, such that

$$\left\| \frac{d}{dt}\left(J^c_{0c_2} \, \dot{\widehat{s}}_x \right) \right\| < \bar{L}_{y_1} \,, \tag{B.49}$$

$$\left\| \frac{d}{dt}\left(J^c_{0c_2} \dot{e}_{x_2} \right) \right\| < \bar{L}_{y_2} \,. \tag{B.50}$$

Then, $\left\| \bar{d}_y \right\| < \bar{L}_{y_1} + \bar{L}_{y_2} + L_{y_2} + L_{y_3}$, again guaranteeing finite-time stabilization of system (B.48) after a time $\bar{T}_2 > 0$. Therefore, the quaternion errors (4.19) and (4.20) tend to I_Q and zero (respectively) asymptotically after a time $max(\bar{T}_1, \bar{T}_2)$.

□